



UNIVERSIDAD de VALLADOLID



ESCUELA de INGENIERÍAS INDUSTRIALES

MÁSTER EN MODELIZACIÓN MATEMÁTICA Y COMPUTACIÓN

PROYECTO FIN DE MÁSTER

**APLICACIÓN WEB PARA EL ANÁLISIS Y LA
VISUALIZACIÓN DE DATOS DE ACTIMETRÍA
APLICADOS AL DIAGNÓSTICO DEL TRASTORNO
POR DÉFICIT DE ATENCIÓN E HIPERACTIVIDAD
EN LA INFANCIA**

Autor(es):

Treceño Fernández, Daniel

Tutor(es):

Alberola López, Carlos

**Laboratorio de procesado de
imagen (ETSIT)**

Martín Martínez, Diego

**Laboratorio de procesado de
imagen (ETSIT)**

SEPTIEMBRE — 2014

Autorización de los tutores

Carlos Alberola López y Diego Martín Martínez, personal docente e investigador del Departamento de Teoría de la Señal, Comunicaciones e Ingeniería Telemática de la Universidad de Valladolid, CERTIFICAN que la presente memoria titulada "Aplicación *web* para el análisis y la visualización de datos de actimetría aplicados al diagnóstico del trastorno por déficit de atención e hiperactividad en la infancia" ha sido realizada por Daniel Treceño Fernández en la Universidad de Valladolid, y que constituye la memoria de su trabajo fin de máster, preceptiva para la finalización de sus estudios en el máster de modelización matemática y computación de la Universidad de Valladolid.

Valladolid, 1 de septiembre 2014.

Fdo:

A handwritten signature in blue ink, appearing to read 'Diego Martín Martínez', with a horizontal line underneath it.

Carlos Alberola López

Diego Martín Martínez

ÍNDICE

1	INTRODUCCIÓN	1
1.1	CONTEXTO	1
1.1.1	<i>TRASTORNO POR DÉFICIT DE ATENCIÓN E HIPERACTIVIDAD</i>	1
1.1.2	<i>ACTIGRAFÍA</i>	4
1.2	MOTIVACIÓN	5
1.3	OBJETIVO DEL PROYECTO	6
1.4	METODOLOGÍA	6
2	PROCESADO Y CLASIFICACIÓN DE SEÑALES ACTIGRÁFICAS	8
2.1	LECTURA DE LOS DATOS	8
2.2	DIEZMADO	10
2.3	CÁLCULO DE LA MÁSCARA DE ACTIVIDAD-DESCANSO	11
2.4	ANÁLISIS ESTADÍSTICO	12
2.5	ANÁLISIS DEL SUEÑO	14
2.6	ANÁLISIS RITMOMÉTRICO (COSINOR)	17
2.7	DINÁMICA SIMBÓLICA	21
2.8	MEDIDA DE LA TENDENCIA CENTRAL (CTM).....	21
2.9	K-NEAREST NEIGHBORS (KNN).....	22
3	INTRODUCCIÓN A LAS TECNOLOGÍAS WEB	25
3.1	HISTORIA	25
3.2	GENERACIONES WEB	27
3.3	¿CÓMO FUNCIONA LA WEB?	28
3.3.1	<i>CLIENTE</i>	29
3.3.2	<i>SERVIDOR</i>	30
3.4	DJANGO	32
3.4.1	<i>ESQUEMA MTV</i>	33
3.4.2	<i>UN PROYECTO EN DJANGO</i>	34
4	DESARROLLO DE LA APLICACIÓN	35
4.1	TOMA DE DECISIONES	35
4.2	ESTRUCTURA DEL PROYECTO.....	36
4.3	CONFIGURACIÓN DEL PROYECTO	39
4.3.1	<i>SETTINGS.PY</i>	39
4.3.2	<i>URLS.PY</i>	42
4.4	DISEÑO Y ARQUITECTURA DE LA PÁGINA WEB	44

4.5	APPLICATION	47
4.5.1	<i>MODELO</i>	48
4.5.2	<i>DISEÑO DE LA INTERFAZ Y PETICIONES AL SERVIDOR</i>	50
4.5.3	<i>VISTA</i>	54
4.6	DOWNLOAD RESULTS	59
4.6.1	<i>DISEÑO DE LA INTERFAZ Y PETICIONES AL SERVIDOR</i>	59
4.6.2	<i>VISTA</i>	61
4.7	CONTACT	62
4.7.1	<i>DISEÑO DE LA INTERFAZ Y PETICIONES AL SERVIDOR</i>	63
4.7.2	<i>VISTA</i>	64
4.8	LOG-IN/LOG-OUT	64
4.8.1	<i>MODELO</i>	64
4.8.2	<i>DISEÑO DE LA INTERFAZ</i>	65
4.8.3	<i>VISTA</i>	66
4.9	OTHER	66
4.9.1	<i>DISEÑO DE LA INTERFAZ</i>	67
4.9.2	<i>VISTA</i>	68
4.10	ADMIN	68
5	CONCLUSIONES Y LÍNEAS FUTURAS	70
5.1	CONCLUSIONES	70
5.2	LÍNEAS FUTURAS	71
6	BIBLIOGRAFÍA	73
7	ANEXO	79

1 INTRODUCCIÓN

Este primer capítulo sirve para fijar el contexto, la motivación del trabajo y los objetivos del proyecto. Con este fin, durante la sección 1.1, se introducen los conceptos fundamentales en actigrafía y TDAH. En la sección 1.2 se justifica brevemente el por qué es necesaria la creación de una aplicación, mientras que en la 1.3 se detallan las características que debe cumplir la aplicación. Finalmente en el punto 1.4 se describe la metodología para la creación de la misma.

1.1 CONTEXTO

El contexto se divide en dos partes; en primer lugar, el trastorno por déficit de atención e hiperactividad, donde se tratará de explicar la situación actual de este trastorno así como los métodos comúnmente utilizados para su diagnóstico. La segunda parte trata de explicar brevemente en qué consiste la actigrafía y cómo puede ayudar ésta en el diagnóstico de dicho trastorno.

1.1.1 TRASTORNO POR DÉFICIT DE ATENCIÓN E HIPERACTIVIDAD

El TDAH es uno de los trastornos más comunes en la niñez, pudiendo continuar hasta la adolescencia o la edad adulta (1). Se estima que lo padece entre un 2% y 16% de la población infantil dependiendo del criterio para su diagnóstico, así como del lugar donde se realiza (2) (por ejemplo, 10% en Carolina del Norte (3), 6,8% en Australia (4), entre un 7 y un 10% en España (5)). Normalmente, este trastorno no se presenta en solitario, sino junto a otros trastornos neurológicos y/o comportamentales como ansiedad, tics, trastornos del estado de ánimo, etc.

Este hecho, unido a la dificultad que sufren los pacientes para llevar una vida normal como consecuencia de los síntomas (movimiento constante, impulsividad, falta de atención, dificultad de concentración, necesidad de cambiar de tarea constantemente, etc.), ha convertido al TDAH en una de las principales dianas de la pediatría/psiquiatría de hoy en día. Pese a ello, no existe un método estándar objetivo para su diagnóstico y, por tanto, su diagnóstico y seguimiento está basado en la evaluación (subjetiva) de los síntomas que presentan los pacientes. En concreto, el protocolo estándar de diagnóstico es el recogido en el DSM-IV (6). El desconocimiento acerca de la etiología del TDAH motiva, en buena medida, la ausencia de un método diagnóstico objetivo (7) (8) (9) (10) (11) (12).

Existen 3 tipos de TDAH dependiendo los de síntomas que se presenten:

- **Predominante hiperactivo-impulsivo:** un paciente es diagnosticado de esta variedad de TDAH si presenta al menos seis síntomas de índole hiperactiva/impulsiva y menos de seis síntomas característicos de inatención conforme al DSM-IV. Dicho de otro modo, los pacientes cuya sintomatología es predominantemente de tipo hiperactivo/impulsivo son diagnosticados de esta variedad de TDAH.
- **Predominantemente inatento:** aquellos pacientes cuyos síntomas más significativos estén asociados a la inatención, son diagnosticados con esta variedad. En concreto, debe presentar, al menos, seis síntomas de inatención y menos de seis síntomas de hiperactividad/impulsividad.
- **Combinación hiperactivo-impulsivo e inatento:** esta variedad se caracteriza porque en el cuadro sintomatológico del paciente no predomina un tipo sobre el otro. Específicamente, debe presentar al menos seis síntomas de cada tipo.

Los métodos para el diagnóstico del TDAH pueden agruparse en:

- Métodos subjetivos
- Métodos objetivos

Métodos subjetivos

Se refiere a todos aquellos métodos que basan su diagnóstico en información subjetiva tal como:

- Informes de comportamiento proporcionados por los padres y los profesores.
- Información sobre el sueño dada por los padres o sus homólogos.

- Entrevistas realizadas a los pacientes

Métodos objetivos

Engloba aquellos métodos cuya información se basa en datos (información objetiva):

- TEST MENTALES como el de Bruch (13) utilizado para la detección de ansiedad.
- ANÁLISIS BIOMÉDICOS tales como el nivel de plomo en el pelo o el nivel de neurotransmisores y receptores de proteínas en sangre (14) (15).
- IMÁGENES MÉDICAS: imágenes de resonancia magnética (MRI) para caracterizar la forma del Putamen y otros núcleos basales (16), tensor de difusión MRI (DT-MRI) para analizar la integridad y la posición de algunas fibras del tracto (17) (18) (19) y PET/SPECT para obtener conjuntamente información anatómica y metabólica (20).
- SEÑALES BIOMÉDICAS: gran parte de los estudios se han centrado en encefalogramas (21) (22) (23) (24) (25) (26) (EEG) y potencial relativo a eventos (27) (28) (ERP). También han sido aplicadas técnicas más innovadoras como la magnetoencefalografía (29).
- POLISOMNOGRAFÍA (PSG): la PSG consiste en el empleo conjunto de una serie de señales biomédicas (encefalogramas, electrooculografía, electrocardiografía, señales respiratorias y pulso-oximetría, etc.(30)), para evaluar el sueño (etapas, calidad, etc.). Actualmente, esta técnica es considerada el patrón oro en la evaluación del sueño; es por ello la más usada para diagnosticar desórdenes de sueño. Varios estudios han analizado los patrones de sueño obtenidos con PSG con dos objetivos:
 1. Demostrar que existen diferencias significativas entre pacientes diagnosticados con TDAH y control cuando duermen (31) (32).
 2. Evaluar el rendimiento del PSG comparado con otros métodos subjetivos (33) (34).
- DETECCIÓN DE MOVIMIENTOS MEDIANTE INFRARROJOS: el uso de infrarrojos para evaluar el movimiento de cabeza y cuello ha sido empleado por Teicher et al. (35) para evaluar pacientes con TDAH. Los autores concluyeron que los movimientos de los pacientes de TDAH eran más frecuentes, cubrían áreas más grandes y eran más lineales y menos complejos que los pacientes que no sufrían este trastorno. Estudios previos (36) (37) revelaron que la combinación de los patrones de movimientos complejos junto con otras variables como la edad o las puntuaciones conseguidas en actividades específicas pueden proporcionar una ayuda para un diagnóstico objetivo y dar medidas precisas para evaluar la respuesta al tratamiento.

- **ACTIGRAFÍA:** es una medida de la actividad realizada por el paciente durante un periodo de tiempo, este método se explica con mayor detenimiento en el siguiente punto.

1.1.2 ACTIGRAFÍA

La actigrafía puede definirse como un método no invasivo que permite, mediante la colocación de un pequeño dispositivo en alguna parte del cuerpo, medir y/o registrar las aceleraciones sufridas sobre un rango dinámico prefijado. Fue usado por primera vez por Schulman and Reisman en 1959, lo hicieron modificando un reloj de forma que las manecillas del reloj mostrasen la aceleración vertical (38) (39). Desde entonces, la actigrafía se ha convertido en una técnica muy útil para la evaluación de la actividad física.

El dispositivo utilizado comúnmente es llamado actígrafo (Figura 1). La forma habitual en la que se presenta puede asemejarse a un reloj de pulsera que suele colocarse en el brazo dominante. Actualmente pueden contar con una gran variedad de sensores como: sensores fotoeléctricos, inclinómetros, acelerómetros en distintos ejes... Consta además de una memoria donde quedan registradas las distintas medidas de los sensores.



Figura 1: Actigraph GT3X

En cuanto al uso de la actigrafía aplicada al diagnóstico de TDAH, puede dividirse en tres modalidades:

- **Actividad diurna:** es una medida de la actividad que se realiza durante el día, periodo en el cual los niños con TDAH tienen una mayor actividad media (40) (41). Teicher et al. (42) mostraron que el porcentaje de tiempo utilizado en el desarrollo de actividades con bajo, medio y alto nivel físico proporciona un mejor indicador de actividad alterada. Niños con TDAH dedican más tiempo diurno a actividades con una mayor intensidad que a actividades más relajadas (37). Un estudio reciente (43) también encuentra diferencias significativas entre los histogramas de niños con TDAH y los de

control, ya que la media y la varianza de la media muestral de los histogramas calculados durante etapas de un minuto a lo largo de las clases (arte, lengua y matemáticas) es más alta en los pacientes con TDAH. En contra de lo esperado (44), no se pudo observar una buena correlación entre la actigrafía y la información subjetiva de la actividad diurna proporcionada, lo cual pone en entredicho la utilidad de la información subjetiva en determinadas circunstancias.

- **Actividad nocturna o sueño:** con la información proporcionada por el actímetro durante el sueño, puede extraerse una gran variedad de parámetros como la latencia, la eficiencia del sueño, etc. mediante el uso de algoritmos (45) (46) (47) (48). La validez de este registro ha sido ampliamente demostrada (31) (34), pero existen algunos problemas para la adaptación de los algoritmos más populares (Sadeh (46) (45), Jean-Luise (47), Cole-Kripke (48), etc.) debido a que se han llevado a cabo con una población específica y actígrafos determinados (49). El porcentaje de actividad también es más alto durante la noche en los pacientes con TDAH.
- **Ritmo biológico circadiano:** los estudios cronobiológicos basados en la actigrafía han detectado que el ritmo circadiano está retrasado en la población adulta afectada por el TDAH (15). Sin embargo, no hay estudios realizados en rangos de edad inferiores.

1.2 MOTIVACIÓN

La principal motivación para la realización de este trabajo es acercar y facilitar a los médicos y psicólogos, el uso de la actigrafía para ayudar a diagnosticar con mayor eficiencia a los pacientes de TDAH. Con este se proyecto se pretende mejorar el acceso de los profesionales a esta tecnología, con el fin de que puedan explorar y experimentar las bondades y los límites de la actigrafía para el diagnóstico y, en la medida de lo posible, contribuir a la objetivación del mismo.

El desarrollo de esta aplicación está basado en los estudios realizados por el Laboratorio de Procesado de Imagen (LPI) de la Universidad de Valladolid (UVa) en esta materia (50) (51). Estos estudios se centran en pacientes en torno a los 6 años de edad con TDAH predominantemente hiperactivo-impulsivo y combinación hiperactivo-impulsivo e inatento.

1.3 OBJETIVO DEL PROYECTO

El objetivo de este proyecto es crear una aplicación web capaz de proporcionar a médicos y psicólogos una herramienta independiente de la plataforma y de alta disponibilidad que permita analizar y gestionar datos actigráficos.

Dicha aplicación contará con:

- Un enlace con una base de datos para el almacenaje y gestión de los archivos actigráficos
- Una batería de algoritmos para el análisis y la extracción de características:
 - Determinación de la máscara de actividad-descanso (50).
 - Selección del canal entre los disponibles en el archivo.
 - Sistema de diezrado (51).
 - Análisis estadístico.
 - Análisis del sueño (45) (46).
 - Análisis cronobiológico (cosinor) (52).
 - Dinámica simbólica (51).
 - CTM (Medida de la tendencia central) (51).
- Un clasificador entrenado previamente para el diagnóstico de pacientes con TDAH con edades cercanas a los 6 años.
- Una herramienta mediante la cual puedan calcularse y descargarse datos de una forma cómoda para el usuario.

1.4 METODOLOGÍA

El proceso seguido para llevar cabo este proyecto puede dividirse en dos partes:

1. La parte referente a la actigrafía:
 - a. Estudio del registro de actigrafía.
 - b. Procesado de la señal actigráfica.
2. La parte referente a la aplicación web:
 - a. Descripción de las tecnologías web.
 - b. Desarrollo de la aplicación.

Por tanto, la primera etapa consiste en el estudio de todos los algoritmos referentes a la actimetría que serán necesarios implementar para la aplicación.

En la segunda etapa se realiza un breve estudio de la evolución web y de las posibles frameworks y lenguajes disponibles para la posterior implementación de la aplicación web.

Finalmente, el diseñada de la aplicación web, así como de cada uno de sus componentes.

2 PROCESADO Y CLASIFICACIÓN DE SEÑALES ACTIGRÁFICAS

Este capítulo detalla los algoritmos y métodos que serán utilizados para la lectura del registro de actigrafía, la extracción de características así como la clasificación de la señal. Cada uno de ellos es analizado en las siguientes sub-secciones.

2.1 LECTURA DE LOS DATOS

Los datos actigráficos son almacenados en un registro dentro del propio actímetro. En el caso que atañe, el registro (Figura 2) se estructura en dos partes:

1. **Cabecera:** contiene toda la información referente al modo de captación de los datos, el intervalo temporal entre cada dato y la fecha de comienzo del registro.
2. **Datos:** los datos almacenados pueden proceder de diferentes sensores colocados en el actígrafo, como acelerómetros en los distintos ejes, sensores fotoeléctricos, inclinómetros, etc. Esta parte del registro contiene las medidas de los mismos.

Debido a la gran cantidad de posibilidades, antes de crear el programa se fijaron una serie de condiciones que deben cumplir el dispositivo y el registro:

- El dispositivo debe pertenecer a la familia de actígrafos GT3X
- El formato del archivo ha de ser como el que se indica en la Figura 2, con extensión *.dat
- Los modos en los que se puede configurar son el 44 y el 45 (Figura 3).

```

----- Data File Created By ActiGraph GT3X ActiLife v4.3.0 Firmware v2.2.0 -----
Serial Number: MAT2A40099713
Start Time 14:00:00
Start Date 06/04/2010
Epoch Period (hh:mm:ss) 00:00:01
Download Time 13:58:12
Download Date 07/04/2010
Current Memory Address: 775854
Current Battery Voltage: 4.02      Mode = 45

```

CABECERA

0	0	0	0	1	139	207	5	0	1	60
18	172	66	0	1	0	44	0	0	1	0
38	8	31	1	1	43	55	55	0	1	14
58	92	37	0	1	17	32	27	1	1	53
86	86	118	0	1	73	12	136	0	1	26
18	131	29	0	2	0	25	0	0	3	8
0	0	0	0	2	0	0	0	0	2	0
8	0	0	0	1	0	0	0	0	3	0
0	0	0	0	2	0	0	0	0	2	0
0	0	0	0	3	0	8	0	0	3	68
0	6	0	0	1	0	5	0	0	2	55

DATOS

Figura 2: Registro generado por un actígrafo GT3X

Mode	Mode in binary	Activity	Dual Activity	3rd Axis	Steps	Heart Rate	Lux	Incline	Total Parameters
0	000000	X							1
1	000001	X			X				2
2	000010	X				X			2
3	000011	X			X	X			3
4	000100	X	X						2
5	000101	X	X		X				3
6	000110	X	X			X			3
44	101100	X	X	X				X	4
45	101101	X	X	X	X			X	5
46	101110	X	X	X		X		X	5
47	101111	X	X	X	X	X		X	6
48	110000	X					X	X	3
49	110001	X			X		X	X	4
50	110010	X				X	X	X	4

Figura 3: Extracto de la tabla 17 del ActiLife 5 - User's Manual

Cumplidos estos requisitos, el algoritmo seguido para la lectura de datos de cada archivo es el siguiente:

1. Abrir el fichero.
2. Almacenar los datos de la cabecera en una cadena de caracteres.
3. Almacenar todos los datos del registro en un único vector.
4. Cerrar el archivo.
5. Extraer los datos de la cabecera:
 - Mode: es el modo con el que se tomaron los datos.
 - Epoch: es el tiempo que transcurre entre registros. Viene en el formato (hh:mm:ss).
 - Time: es la hora en la que comenzó el registro (hh:mm:ss).

- Date: es la fecha en que comenzó el registro (dd/mm/yyyy).
6. Si el modo* se corresponde con alguno de los implementados:
 1. Se fijan los siguientes parámetros en función del modo:
 - nChannels: Número de canales
 - Channels_names: Nombre de los canales
 - VarVectorModule**: El número de canales con el que se calculará el vector módulo. Puede ser mayor o igual a cero. En caso de ser cero no se calculará.
 - N_channels: El número de canales total contando el vector módulo
 2. En función de los parámetros anteriores, se almacenan en forma de lista todos los canales, de manera que Channel[0] correspondería al primer canal.
 3. Se calculan otras características útiles:
 - End_time: hora a la que se finaliza el registro.
 - Total_time: tiempo total de registro.
 7. Si no, se retorna un error.

**Si se necesita implementar otro registro, debe añadirse un nuevo modo en el punto 6 tal y como se muestra en el siguiente ejemplo:*

```
if Mode == 44:
    nChannels = 4
    Channels_names = ['x Axis', 'y Axis', 'z Axis', 'Inclinometer', 'Vector Module']
    VarVectorModule = 3
    N_channels = 5
```

***El Vector Module es un array de datos adicional que se crea con los canales de actividad. Siendo x, y, z los canales de actividad y v el Vector Module, éste se calcula como*

$$v_i = \sqrt{x_i^2 + y_i^2 + z_i^2}, 0 \leq i \leq n, \text{ donde } n \text{ es la longitud del canal}$$

2.2 DIEZMADO

El diezmado aplicado a este tipo de señales produce una mejora de la información de la actividad que se halla oculta por la gran cantidad de ceros que existe en la señal original (45) (46) (47) (48). La forma de calcular la señal diezmada (51) es la siguiente:

$$x_{dec}[k] = \frac{1}{M} \sum_{i=0}^{M-1} x[k \cdot M + i], 0 \leq k \leq \frac{N}{M} \quad [1]$$

Donde:

x_{dec} → Señal diezmada
 x → Señal original
 M → Diezmo
 N → Longitud de la señal original

2.3 CÁLCULO DE LA MÁSCARA DE ACTIVIDAD-DESCANSO

La detección de los intervalos de actividad y descanso se ha realizado mediante el método propuesto por Martín et al (50). Este método proporciona una señal (máscara) que toma valor nulo en los instantes de descanso, y uno en los instantes de actividad. Dicho método sigue los siguientes pasos:

Se normaliza la señal:

$$x_{unit}[n] = \frac{x[n]}{\max\{x[n], n = 1, 2, \dots, N\}} \quad [2]$$

Donde:

x_{unit} → Señal normalizada
 x → Señal actigráfica de 24 horas
 N → Longitud de la señal

Se realiza la convolución de la señal normalizada con una ventana triangular invertida de 30 minutos (1800 segundos).

$$x_{av}[n] = x_{unit}[n] * h_{it}[n] \quad [3]$$

Donde:

x_{av} → Señal filtrada
 x_{unit} → Señal unitaria
 h_{it} → Ventana triangular

Para la identificación de la actividad-descanso se umbraliza la señal filtrada:

$$m_A[n] = \begin{cases} 0 & \text{si } x_{av} < th \\ 1 & \text{en otro caso} \end{cases} \quad [4]$$

Donde:

m_A → Máscara de actividad
 x_{av} → Señal filtrada
 th → Umbral de 0,025

Finalmente, se aplica un método iterativo para el cálculo de m_A que finaliza cuando no existen diferencias entre dos m_A consecutivas. El proceso consiste en calcular una nueva señal:

$$x_{av,iterative}[n] = x_{av}[n] \times m_A[n] \quad [5]$$

y a partir de esta, recalcular nuevamente m_A mediante [4].

2.4 ANÁLISIS ESTADÍSTICO

El análisis estadístico trata de "caracterizar" o "resumir" los datos generados a partir de la actigrafía. Se compone por algunos de los parámetros estadísticos más comunes:

Media aritmética: Es el promedio de un conjunto de datos cuantitativos.

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n a_i \quad [6]$$

Donde:

\bar{x} → Media aritmética
 a_i → Registro de datos diezmado
 n → Longitud de la señal de datos

Moda: Es el valor con mayor frecuencia en una distribución de datos.

Desviación estándar: Se define como la raíz cuadrada de la varianza.

$$\sigma = s = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2} \quad [7]$$

Donde:

σ → Desviación estándar

x_i → Registro de datos diezmado
 n → Longitud de la señal de datos
 \bar{x} → Media aritmética

Rango intercuartílico: Es la diferencia entre el tercer y el primer cuartil. La forma en la que se aplica es la creada por Hoaglin et al. (53):

$$IQR = Q_3 - Q_1 = x_{\left(\frac{3*n+1}{4}\right)} - x_{\left(\frac{n+3}{4}\right)} \quad [8]$$

Donde:

x → Es la señal del registro ordenada de menor a mayor
 n → Longitud de la señal de datos

En caso de que el índice no sea un entero, se realiza una interpolación lineal entre los 2 valores más cercanos. Suponiendo que el índice resultante sea:

$$ind = [ind] + \{ind\} \quad [9]$$

Donde:

ind → Número real
 $[ind]$ → Parte entera
 $\{ind\}$ → Parte fraccional

Entonces el valor del cuartil sería:

$$x_{ind} = x_{[ind]} + (x_{[ind]+1} - x_{[ind]}) * \{ind\} \quad [10]$$

Donde:

x → Es la señal de datos ordenada de menor a mayor

Asimetría estadística: Es una medida de asimetría de la distribución; la ecuación para una señal discreta puede escribirse de la forma (54):

$$skew = \frac{\mu_3}{\sigma^3} = \frac{\frac{1}{n} \sum (X_i - \bar{X})^3 * n_i}{\left(\frac{1}{n} \sum (X_i - \bar{X})^2 * n_i\right)^{3/2}} \quad [11]$$

Donde:

μ_3 → Tercer momento en torno a la media
 σ → Desviación estándar
 n → Total de valores distintos de la señal

X_i	→	<i>Cada uno de los valores</i>
\bar{X}	→	<i>Media de la muestra</i>
n_i	→	<i>Frecuencia de cada valor</i>

Curtosis: Es la medida que determina el grado de concentración que presentan los valores en la región central de la distribución; la ecuación para una señal discreta puede escribirse de la forma (54):

$$kurtosis = \frac{\mu_4}{\sigma^4} - 3 = \frac{\frac{1}{n} \sum (X_i - \bar{X})^4 * n_i}{\left(\frac{1}{n} \sum (X_i - \bar{X})^2 * n_i \right)^2} - 3 \quad [12]$$

Donde:

μ_4	→	<i>Cuarto momento en torno a la media</i>
σ	→	<i>Desviación estándar</i>
n	→	<i>Total de valores distintos de la señal</i>
X_i	→	<i>Cada uno de los valores</i>
\bar{X}	→	<i>Media de la muestra</i>
n_i	→	<i>Frecuencia de cada valor</i>

Coefficiente de variación: Es una medida de dispersión que establece una relación entre el tamaño de la media y la variabilidad de la variable:

$$c_v = \frac{\sigma}{\bar{x}} \quad [13]$$

Donde:

c_v	→	<i>Coefficiente de variación</i>
σ	→	<i>Desviación estándar</i>
\bar{x}	→	<i>Media aritmética</i>

2.5 ANÁLISIS DEL SUEÑO

Tomando sólo la parte del registro considerada como descanso, se emplea el algoritmo de Sadeh et al. (45) (46) para realizar el análisis del sueño.

El algoritmo está formado por las siguientes etapas:

1. Diezmado de la señal. La señal original es diezmada en etapas de 60 segundos (sección 2.2) obteniendo la señal x_{dec}

2. Se multiplica la señal x_{dec} por una constante $k = 2,54$. Esta constante se debe a que el dispositivo usado en el algoritmo original era distinto, y para paliar esta diferencia, la señal debe multiplicarse por esta variable. Para su estimación, se han empleado un conjunto de señales que se han analizado mediante el software provisto por Actigraph; el valor de k se ha estimado como aquél que minimiza el error (en mínimos cuadrados) entre los parámetros estimados con la implementación en MATLAB y aquéllos obtenidos mediante el software de Actigraph
3. Se calculan los siguientes parámetros a partir de x_{dec} :

- NZW_i : El número de minutos con actividad cero. Es el resultado de la suma de los ceros que hay en la posición actual, 5 minutos antes y 5 minutos después.

$$NZW_i = \sum (x_{dec} [i - 5 : i + 5] == 0) \quad [14]$$

- NTL_i : El número de minutos con baja actividad.

$$NTL_i = \sum (x_{dec} [i - 5 : i + 5] < 100) - NZW_i \quad [15]$$

- NTH_i : El número de minutos con actividad alta.

$$NTH_i = \sum (x_{dec} [i - 5 : i + 5] \geq 100) \quad [16]$$

- $S5_i$: La desviación estandar de una ventana de los siguientes 5 minutos.

$$S5_i = std(x_{dec} [i : i + 5]) \quad [17]$$

- ML_i : El nivel de actividad medio entorno al anterior y al actual minuto.

$$ML_i = \frac{1}{2} \sum (x_{dec} [i - 1 : i]) \quad [18]$$

- $LW4_i$: La actividad más baja en los inmediatos 4 minutos.

$$LW4_i = \min(x_{dec} [i : i + 5]) \quad [19]$$

4. Con los parámetros anteriores, se calcula:

- PQS : Es la probabilidad de estar dormido.

$$PQS = 15.94 + 3.223 \cdot NZW + 2.138 \cdot NTL + 1.1036 \cdot NTH + 0.0466 \cdot S5 + 0.00292 \cdot ML + 0.0106 \cdot LW4 \quad [20]$$

- PAS : Es la probabilidad de estar en una etapa activa de sueño.

$$PAS = 5.134 + 1.696 \cdot NZW + 2.062 \cdot NTL + 0.9568 \cdot NTH + 0.0585 \cdot S5 \\ + 0.00556 \cdot ML + 0.0105 \cdot LW4 \quad [21]$$

- *PAW*: Es la probabilidad de vigilia.

$$PAW = -25.638 + 1.714 \cdot NZW + 3.0168 \cdot NTL + 4.064 \cdot NTH + 1.1066 \cdot S5 \\ + 0.0386 \cdot ML + 0.016 \cdot LW4 \quad [22]$$

5. Se calcula la máscara que determina cómo ha sido el descanso minuto a minuto. Se le denomina *PS*. Es 1 cuando la probabilidad de que esté durmiendo es alta.

$$PS = ((PQS \geq PAW) \text{ ó } (PAS \geq PAW)) \quad [23]$$

6. Finalmente se calculan las siguientes marcas:

- **Tiempo total dormido (*TTD*)**: Se da en minutos.

$$TTD = \sum (PS == 1) \quad [24]$$

- **Tiempo despierto después de dormirse (*TDDD*)**: Es el total de minutos que ha estado despierto una vez que se durmió la primera vez.

$$TDDD = \sum (PS == 0) \quad [25]$$

- **Número de despertares (*ND*)**: El número de veces que la actividad ha sido alta. Se evalúa el número de veces que *PS* cambia de 0 a 1.
- **Duración media del despertar (*DMD*)**: Es la media de la duración de cada despertar.

$$DMD = \frac{TDDD}{ND} \quad [26]$$

- **Actividad total (*AT*)**:

$$AT = \sum x_{dec} \quad [27]$$

- **Intervalo de sueño más largo (*ISL*)**: Es el intervalo más largo de sueño. Se determina contando el número de minutos por cada intervalo de sueño y tomando el máximo.
- **Eficiencia del sueño (*ES*)**:

$$ES = \frac{TTD}{length(PS) + 10} \cdot 100 \quad [28]$$

- **Porcentaje de sueño sin actividad (PSSA):** Es el porcentaje de tiempo que ha estado dormido con actividad cero.

$$PSSA = \frac{\sum(PQS[PS] \geq PAS[PS])}{length(PS)} \cdot 100 \quad [29]$$

2.6 ANÁLISIS RITMOMÉTRICO (COSINOR)

Existe un gran número de modelos matemáticos que han sido utilizados para asociar funciones periódicas con ritmos biológicos. Una de las técnicas más comunes es el "cosinor rhythmometry", que fue desarrollado por Harberg et al. (55). Más tarde, Nelson et al. (52) proporcionaron una revisión más detallada de las técnicas de cálculo y análisis, que serán las utilizadas en este proyecto.

El algoritmo trata de aproximar una curva sinusoidal [30] a una serie de datos obtenidos a lo largo del tiempo (el periodo suele ser de 24 horas) mediante mínimos cuadrados [31].

$$f = M + A \cdot \cos(\omega \cdot t + \phi) \quad [30]$$

Donde:

f	→	Curva
M	→	Mesor o nivel medio
A	→	Amplitud
ω	→	Pulsación
t	→	Tiempo
ϕ	→	Fase

$$\chi^2 = \sum (y - f)^2 \quad [31]$$

Donde:

χ^2	→	Error cuadrático
y	→	Señal
f	→	Curva que se desea ajustar

Sabiendo que cualquier combinación lineal de una serie de ondas sinodales tienen el mismo periodo pero están desfasadas, es también una onda sinodal del mismo periodo pero con un desplazamiento de fase diferente:

$$f = M + \sqrt{\beta^2 + \gamma^2} \cdot \cos\left(\omega \cdot t + \arctan\left(\frac{\gamma}{\beta}\right)\right) = M + \beta \cdot \cos(\omega \cdot t) + \gamma \cdot \sin(\omega \cdot t) \quad [32]$$

Se toma la última forma de [31] en [32].

$$\chi^2 = \sum \left(y - (M + \beta \cdot \cos(\omega \cdot t) + \gamma \cdot \sin(\omega \cdot t)) \right)^2 \quad [33]$$

Se plantea el sistema.

$$\begin{cases} \frac{\partial \chi^2}{\partial M} = 0 \\ \frac{\partial \chi^2}{\partial \beta} = 0 \\ \frac{\partial \chi^2}{\partial \gamma} = 0 \end{cases} \quad [34]$$

Una vez obtenidos M , β , γ , se sustituye en [32] para calcular los parámetros restantes.

La acrofase viene determinada por ϕ , ya que es la fase donde se produce el máximo valor del ritmo biológico, o dicho de otra manera, donde la amplitud es máxima. Se expresa en horas.

$$Acrofase = 24 \cdot \frac{\phi}{2 \cdot \pi} \quad [35]$$

El siguiente paso consiste en calcular los intervalos de confianza (CI) para el mesor, la amplitud y la acrofase. Para ello es necesario establecer el parámetro α que determinará la probabilidad o el nivel de confianza (LC) del intervalo.

$$LC(\%) = 100 - \alpha \quad [36]$$

Se calcula el error residual (χ^2) [33] y se hace una estimación de la desviación (σ).

$$\sigma = \frac{\sqrt{\chi^2}}{n - 3} \quad [37]$$

Donde:

$n \rightarrow$ Es la longitud de y

Para cálculos siguientes es necesario obtener también las siguientes varianzas y covarianzas:

$$\begin{aligned}
x &= \frac{1}{n} \cdot \sum (\cos(\omega \cdot t) - \mu(\cos(\omega \cdot t)))^2 \\
z &= \frac{1}{n} \cdot \sum (\sin(\omega \cdot t) - \mu(\sin(\omega \cdot t)))^2
\end{aligned}
\tag{38}$$

$$xz = \frac{1}{n} \cdot \sum ((\cos(\omega \cdot t) - \mu(\cos(\omega \cdot t))) \cdot (\sin(\omega \cdot t) - \mu(\sin(\omega \cdot t))))$$

Donde:

$$\mu(a) \rightarrow \text{Es la media del vector } a$$

El intervalo de confianza del mesor viene determinado por el máximo y el mínimo valor de CI_M .

$$\begin{aligned}
CI_M &= T^{-1}\left(\frac{100 - \alpha}{2 \cdot 100}, n - 3\right) \cdot \sigma^2 \\
&\cdot \sqrt{\frac{\sum(\cos(\omega \cdot t)^2) \cdot \sum(\sin(\omega \cdot t)^2) - \sum(\cos(\omega \cdot t)^2 \cdot \sin(\omega \cdot t)^2)}{n^3 \cdot (x \cdot z - xz^2)}}
\end{aligned}
\tag{39}$$

Donde:

$T^{-1}(p, v) \rightarrow$ Es la función de distribución acumulativa t de Student inversa, donde " p " es la probabilidad y " v " los grados de libertad.

Los intervalos de confianza de la amplitud y de la acrofase son calculados a partir de la elipse de confianza que se determina utilizando las siguientes expresiones:

$$\begin{aligned}
a1 &= x \\
a2 &= 2 \cdot xz \\
a3 &= z \\
a4 &= -2 \cdot x \cdot \beta - 2 \cdot xz \cdot \gamma \\
a5 &= -2 \cdot xz \cdot \beta - 2 \cdot z \cdot \gamma \\
a6 &= x \cdot \beta^2 + 2 \cdot xz \cdot \beta \cdot \gamma + x \cdot \gamma^2 - \frac{2}{n} \cdot \sigma^2 \cdot F^{-1}\left(\frac{100 - \alpha}{100}, 2, n - 3\right) \\
g_{max} &= -\frac{2 \cdot a1 \cdot a5 - a4 \cdot a3}{4 \cdot a1 \cdot a3 - a2^2}
\end{aligned}
\tag{40}$$

Donde:

$F^{-1}(p, v1, v2) \rightarrow$ Es la función de distribución acumulativa inversa, donde " p " es la probabilidad, " $v1$ " los grados de libertad del numerador y " $v2$ " los grados de libertad del denominador.

La elipse viene formada por los siguientes puntos:

$$\gamma_{CI} = [g_{max} - A \cdot 2 : \frac{A}{1000} : g_{max} + A \cdot 2]$$

$$\beta_{CI1} = \frac{-(a2 \cdot \gamma_{CI} + a4) + \sqrt{(a2 \cdot \gamma_{CI} + a4)^2 - 4 \cdot A \cdot (a3 \cdot \gamma_{CI}^2 + a5 \cdot \gamma_{CI} + a6)}}{2 \cdot A} \quad [41]$$

$$\beta_{CI2} = \frac{-(a2 \cdot \gamma_{CI} + a4) - \sqrt{(a2 \cdot \gamma_{CI} + a4)^2 - 4 \cdot A \cdot (a3 \cdot \gamma_{CI}^2 + a5 \cdot \gamma_{CI} + a6)}}{2 \cdot A}$$

Donde:

A → Es la amplitud
 $[LI: S: LU]$ → Es el vector que comienza en LI, finaliza en LU y el paso entre cada número es de S

Intervalo de confianza de la amplitud, viene determinado por el máximo y el mínimo de los siguientes vectores.

$$CI_{A1} = \sqrt{\beta_{CI1}^2 + \gamma_{CI}^2}$$

$$CI_{A2} = \sqrt{\beta_{CI2}^2 + \gamma_{CI}^2} \quad [42]$$

El intervalo de confianza para la acrofase se consigue calculando el mínimo y el máximo del siguiente vector:

$$CI_{Acrofase} = \left[\text{atan}\left(\frac{\gamma_{CI}}{\beta_{CI1}}\right), \text{atan}\left(\frac{\gamma_{CI}}{\beta_{CI2}}\right) \right] \quad [43]$$

Por último se calcula el test de amplitud cero:

$$p_{value} = fd\left(n \cdot \frac{x \cdot \beta^2 + z \cdot \gamma^2 + xz \cdot \beta \cdot \gamma}{2 \cdot \sigma^2}, 2, n - 3\right) \quad [44]$$

Donde:

$fd(val, v1, v2)$ → Es la función de densidad F, "v1" los grados de libertad del numerador y "v2" los grados de libertad del denominador.

2.7 DINÁMICA SIMBÓLICA

La dinámica simbólica (SD) está basada en el cálculo de la probabilidad de ocurrencia de una palabra en una versión cuantificada $c[n]$ de la señal original $x[n]$ (56) (57). La longitud del alfabeto N_A y el número de símbolos por palabra ℓ son parámetros de esta técnica: $SD(N_A, \ell)$. Al implementar el algoritmo, se establece una longitud máxima $N_A = 36$, donde el alfabeto se compone de forma alfanumérica $A = \{Sym_1, Sym_2, \dots, Sym_{35}\} = \{0, 1, \dots, 9, a, b, \dots, z\}$

La señal cuantificada se construye mediante la fijación de umbrales utilizando los percentiles p_i :

$$c[n] = \begin{cases} Sym_1 & \text{si } x[n] \leq p_{\frac{100}{N_A}} \\ Sym_2 & \text{si } p_{\frac{100}{N_A}} < x[n] \leq p_{2 \cdot \frac{100}{N_A}} \\ \vdots & \vdots \\ Sym_{N_A-1} & \text{si } p_{(N_A-2) \cdot \frac{100}{N_A}} < x[n] \leq p_{(N_A-1) \cdot \frac{100}{N_A}} \\ Sym_{N_A} & \text{En otro caso} \end{cases} \quad [45]$$

La caracterización de la cadena de símbolos viene dada por la distribución de probabilidad de las palabras $w_\ell[k]$.

$$w_\ell[k] = \begin{bmatrix} c[k(\ell - \tau) + \tau] \\ c[k(\ell - \tau) + \tau - 1] \\ \vdots \\ c[k(\ell - \tau) + \tau - (\ell - 1)] \end{bmatrix} \quad [46]$$

Donde τ es el solapamiento entre palabras, esto significa que $\tau = 0, 1, 2, \dots < \ell$

2.8 MEDIDA DE LA TENDENCIA CENTRAL (CTM)

La medida de la tendencia central es una característica no-lineal basada en diferencias de segundo orden. Su cálculo se realiza siguiendo estos pasos:

Dado el conjunto de datos actigráficos

$$x[n], \text{ donde } n = 1, 2, \dots, N$$

Se calculan las diferencias de segundo orden de la señal para el eje de las abscisas y de las ordenadas:

Eje abscisas: $x[n] - x[n - 1]$

Eje ordenadas: $x[n - 1] - x[n - 2]$

El CTM se obtiene calculando la probabilidad de que los puntos estén dentro un círculo de radio ρ situado en el origen de coordenadas:

$$CTM = \frac{1}{N - 2} \sum_{i=1}^{N-2} \delta_i \quad [47]$$

Donde:

$$\delta_i = \begin{cases} 1 & \text{if } \sqrt{(x[n + 2] - x[n + 1])^2 + (y[n + 1] - y[n])^2} \leq \rho \\ 0 & \text{en otro caso} \end{cases} \quad [48]$$

2.9 K-NEAREST NEIGHBORS (KNN)

La intuición intrínseca al clasificador de los k vecinos más cercanos es bastante directa. Las muestras dadas se clasifican en función de los vecinos que se encuentren a la distancia más cercana, y estos son el conjunto de entrenamiento.

Para este caso el espacio definido será bidimensional y la distancia usada será la euclídea. Se parte de un punto $q(x, y)$ que se desea clasificar utilizando un conjunto de entrenamiento $s = \{c_{1,1}, c_{1,2}, \dots, c_{1,n}, c_{2,1}, c_{2,2}, \dots, c_{2,m}\}$ donde $c_{1,i}$ y $c_{2,j}$ son las diferentes clases a las que puede pertenecer q y $N_s = n + m$ es el tamaño del conjunto de entrenamiento. El algoritmo utilizado puede dividirse en dos pasos:

1. Calcular los k vecinos más próximos.
2. Determinar a qué clase pertenece el punto q utilizando los k -vecinos más próximos

Para el primer paso:

1. Se calculan todas las distancias euclídeas entre el punto y cada uno de los puntos del conjunto de entrenamiento.

$$d_i(q, s_i, c) = \sqrt{(c_i - q)^2} \text{ con } i = 1, 2, \dots, N_s \quad [49]$$

Donde:

- d → Distancia euclídea
- q → Punto que se desea clasificar
- s_i → Cada uno de los puntos del conjunto
- c → La clase a la que pertenece $c=1,2$

2. Se ordena de menor a mayor.
3. Se toman los k vecinos más cercanos.

$$KNN = d_sort_i(q, s_i, c) \text{ con } i = 1, 2, \dots, K \quad [50]$$

Segundo paso:

1. Se realiza el conteo de los vecinos más cercanos en función de su clase. La clase que más vecinos cercanos tenga es la asignada al punto q .

$$N_{c=j} = \text{lenght}(d_sort_f(q, s_i, j)) \text{ con } f = 1, 2, \dots, K; j = 1, 2 \quad [51]$$

$$class = \begin{cases} 1 & N_{c=1} > N_{c=2} \\ 2 & N_{c=1} < N_{c=2} \\ 1 \text{ ó } 2 & N_{c=1} = N_{c=2} \end{cases} \quad [52]$$

- a. Si el número de vecinos es el mismo, se ponderan las medias y se le asigna la clase que menor media ponderada tenga.

$$class = \begin{cases} 1 & \frac{1}{N_1} \sum d_sort_f(q, s_i, 1) > \frac{1}{N_2} \sum d_sort_i(q, s_i, 2) \\ 2 & \frac{1}{N_1} \sum d_sort_f(q, s_i, 1) < \frac{1}{N_2} \sum d_sort_i(q, s_i, 2) \\ 1 \text{ ó } 2 & \frac{1}{N_1} \sum d_sort_f(q, s_i, 1) < \frac{1}{N_2} \sum d_sort_i(q, s_i, 2) \end{cases} \quad [53]$$

- i. Si esto también coincide, se toma la clase del primer vecino más cercano. En este caso se retornaría el mismo resultado que si k fuese 1.

$$class = c \text{ de } d_sort_1(q, s_i, c) \quad [54]$$

Las características utilizadas para este clasificador (58) son:

- $CTM(\rho = 45)$ del registro diario completo con un diezmado de 1 segundo en el eje Z .

- $CTM(\rho = 22)$ del registro de actividad con un diezmado de 30 segundos en vector módulo.

3 INTRODUCCIÓN A LAS TECNOLOGÍAS WEB

En este capítulo se presenta una pequeña introducción a las tecnologías web, haciendo hincapié en algunas personas y hechos destacables. Además, se muestra de forma básica la arquitectura cliente/servidor, así como los lenguajes que utilizan y sus tendencias

3.1 HISTORIA

La red de información mundial (WWW - World Wide Web) y su increíble desarrollo ha supuesto que en los últimos años se incremente el uso de este medio para la difusión de información, o la creación de servicios, hasta el punto que cualquier empresa o particular recurra a esta tecnología diariamente. Pero, ¿cómo empezó todo?

Se considera que la primera idea de Web fue propuesta por Vannevar Bush en 1945 con el artículo "As We May Think" en The Atlantic Monthly (59). Trata sobre un dispositivo electro-mecánico con una interfaz capaz de crear y seguir enlaces entre distintos documentos almacenados en microfichas llamado MEMEX ("MEMory" and "EXtender"). Este dispositivo podría asemejarse a lo que hoy se conoce como hipertexto, término acuñado por Ted Nelson en 1965 (60).

A partir de aquí comienzan a desarrollarse diversos sistemas de hipertexto:

- HES - Hypertext Editing System (1967)
- NLS - On-line System (1968)
- FRESS - File Retrieval and Editing System (1969)

Fue la segmentación entre hardware y software lo que en 1980 llevó Tim Berners-Lee, considerado el padre de la Web, a proponer al CERN (Conseil Européen pour la Recherche Nucléaire), lugar en el que trabajaba, el proyecto ENQUIRE (Enquire Within Upon Everything). Este proyecto debía ser compatible entre diferentes redes, formatos de disco, formatos de información, codificación de caracteres, etc. con el fin de poder transferir información entre sistemas diferentes. A grandes rasgos, el programa permitía crear enlaces entre nodos, donde cada nodo posee un título, un tipo y una lista de enlaces.



Figura 4: Tim Berners-Lee (2012)

En 1990 Tim Berners-Lee comienza a desarrollar un editor y navegador de hipertexto, elige WorldWideWeb como nombre. En ese mismo año también se instala el primer servidor web y se publica la primera página web.

Las especificaciones dadas por él sobre las URIs (Uniform Resource Identifier), HTTP (Hypertext Transfer Protocol) y HTML (HyperText Markup Language) fueron clave para la propagación de las tecnologías web.

Un año después, en 1991, comienza la difusión de esta tecnología por internet y el 30 de abril el CERN presenta de forma pública la tecnología WWW, poniéndola a disposición de cualquier persona sin la necesidad de pagar ninguna licencia o canon.

A partir de este momento comenzó una rápida evolución de la Web que podría agruparse mediante el uso de generaciones.

3.2 GENERACIONES WEB

La web ha ido evolucionando desde la información almacenada en páginas estáticas hacia los sistemas de generación dinámica de las páginas web actuales. Cabe dividir dicha evolución en tres generaciones (61):

Web 1.0 es una generación donde predominan las páginas HTML escritas a mano y su carácter es meramente divulgativo. Algunas características son (62):

- Las páginas eran totalmente estáticas. Los formularios HTML eran enviados mediante un servidor de correo electrónico y el contenido gráfico se limitaba habitualmente a imágenes GIF (Graphics Interchange Format).
- Los sitios no eran interactivos: un visitante de la página web no podía contribuir a la misma.



Figura 5: Ilustración publicada por Aysoon sobre las diferencias entre la Web 1.0 y la Web 2.0

Web 2.0, en esta generación se incorpora el concepto de generación dinámica de documentos HTML (63). El comienzo viene marcado por la evolución de las tecnologías, ya que los ordenadores son más potentes y las conexiones más rápidas. Esto permite una gran mejora en los servicios ofrecidos, lo cual lleva a hacer de las webs una gran herramienta

colaborativa. La interactividad crece gracias a los foros, blogs, wikis, servicios de alojamiento de vídeos e incluso redes sociales, permitiendo compartir el conocimiento y la información.

Web 3.0, esta nueva evolución tiene una orientación diferente a la Web 2.0. Es la web inteligente (64) o la web de las búsquedas semánticas pero también es la web de la nube, de las aplicaciones y de los multidispositivos. Trata de aprovechar los servicios on-line para prestar servicios al usuario y eliminar la necesidad de uso de ordenadores potentes con sistemas operativos complejos.

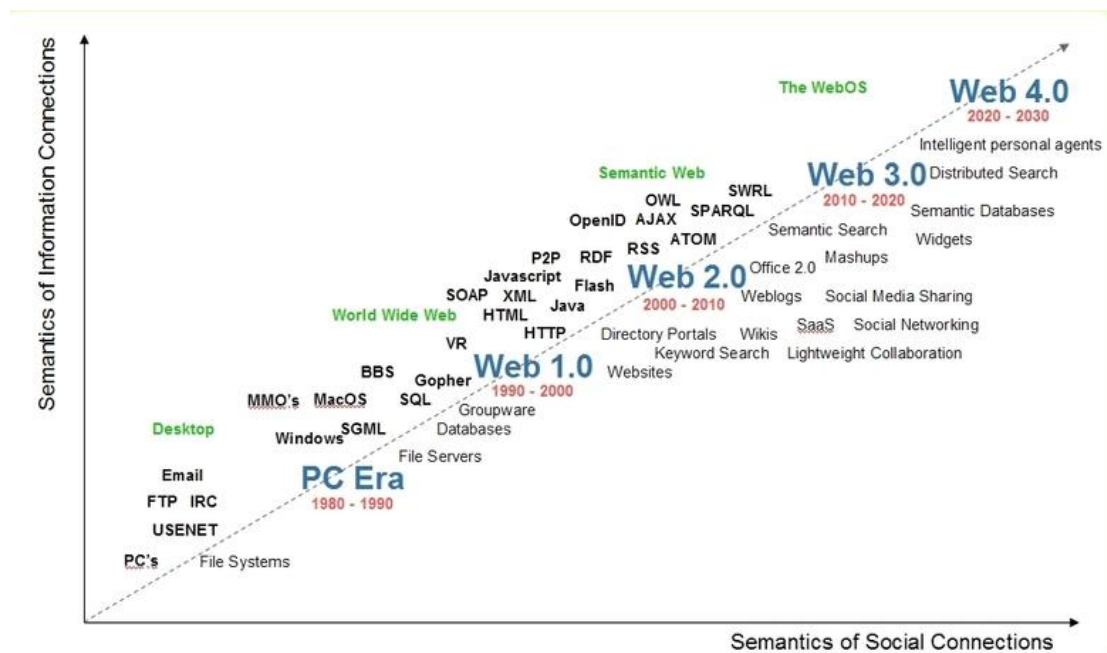


Figura 6: de las tecnologías web - Radar Networks & Nova Spivack, 2007

3.3 ¿CÓMO FUNCIONA LA WEB?

El funcionamiento básico de una página web puede definirse como un tipo específico de aplicación cliente-servidor, donde mediante un protocolo muy simple (*HTTP*), un *cliente* realiza una petición (normalmente mediante un navegador web) a un *servidor web*, el cual la procesa y envía una respuesta (Figura 7).

La forma en que se comunica está totalmente estandarizada por lo que no se va a detallar. Sin embargo, las partes del cliente y del servidor se presentarán de forma algo más detenida.

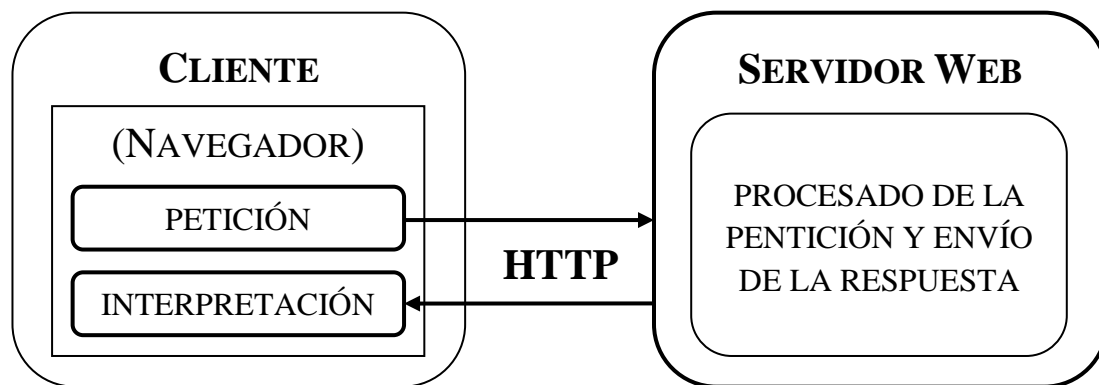


Figura 7: Esquema básico cliente/servidor web

3.3.1 CLIENTE

Habitualmente el cliente web es un programa con el que el usuario interactúa para solicitar al servidor web una información. Este programa es normalmente llamado navegador web y su misión es realizar la petición al servidor e interpretar su respuesta. Las tecnologías que se suelen emplear para su programación son:

- HTML (HyperText Markup Language)
- CSS (Cascading Style Sheets)
- JavaScript (JS) y otros lenguajes del tipo script como puede ser VBScript (Visual Basic Script).
- Applets de JAVA
- Otras tecnologías que requieren la existencia de un plug-in en el navegador: Macromedia Flash, Macromedia Shockwave, Adobe Acrobat Reader, etc.

Una de las combinaciones más populares actualmente es el uso de HTML5, CSS3 y JavaScript. Ésta cuenta con un soporte multimedia completo, API 2D nativa y otras mejoras que pueden acabar eliminando el uso de plug-ins externos en las páginas web. Uno de sus puntos fuertes es el apoyo recibido por grandes compañías como Google, Microsoft y Apple, así como el soporte nativo de la mayoría de los navegadores en sus últimas versiones como Chrome 36, Firefox 30, Internet Explorer 11, Safari 8.0 u Opera 22.

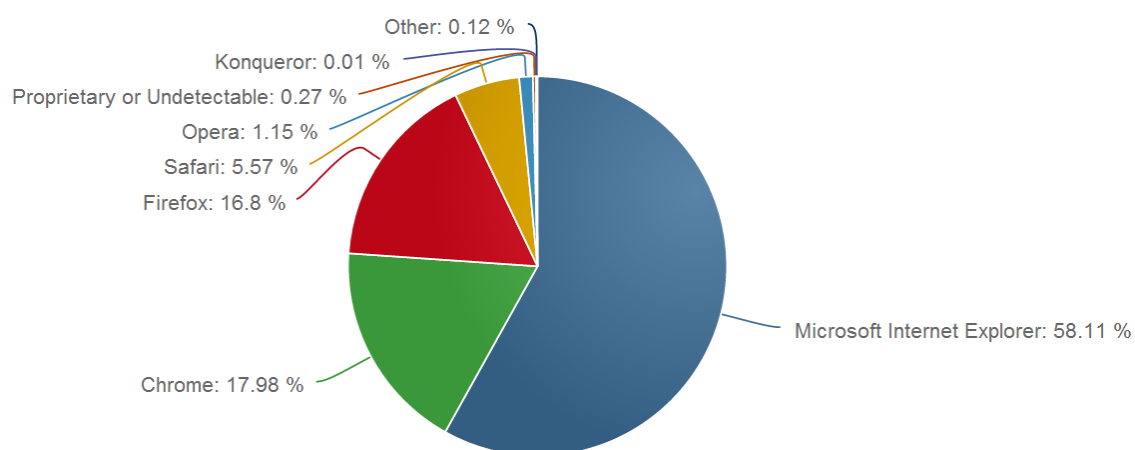


Figura 8: Cuota de mercado en navegadores de escritorio en 2013-2014 según www.netmarketshare.com



Figura 9: Puntuación relativa a la compatibilidad y al soporte de HTML5, CSS3 y otros elementos por navegador según www.html5test.com

La Figura 8 y la Figura 9 llevan a deducir que no todos los navegadores son iguales, y así es, cada que cada uno de ellos cuenta con su propio motor que se encarga de procesar la respuesta enviada por el servidor. Esto hace que las páginas puedan tener un aspecto u otro en función del navegador en el que se ejecuten.

3.3.2 SERVIDOR

El servidor es un programa que está continuamente esperando las solicitudes de conexión mediante el protocolo HTTP por parte de los clientes web. Habitualmente las webs dentro del servidor están formadas por:

- Páginas estáticas que siempre muestran el mismo contenido.

- Recursos adicionales como imágenes, documentos o vídeos que pueden emplearse dentro de páginas o estar disponible para ser descargados por el cliente
- Programas o scripts que son ejecutados por el servidor web cuando el cliente realiza la petición de alguna página. Se refiere a programas escritos comúnmente en PHP (Personal Home Page), ASP (Active Server Pages), Perl, Python ó Ruby. Suelen estar basados en la tecnología CGI (Common Gateway Interface) y contar con la posibilidad de acceder a una base de datos.

El programa CGI puede estar escrito en cualquier lenguaje que genere un fichero ejecutable, pero como CGI recibe los parámetros en forma de texto, uno de los primeros lenguajes utilizados con este fin fue Perl ya que contaba con un gran motor de expresiones regulares.

Pero el uso de CGI resultaba algo complejo y laborioso, ya que el protocolo HTTP no almacena el estado entre una conexión y la siguiente, por lo que el programador tenía que encargarse de conservarlo.

Esto era así hasta que el 8 de junio de 1995 se publicó la primera versión del lenguaje PHP propuesta por Rasmus Lerdorf y más tarde, en diciembre de 1995, Microsoft creó ASP, un add-on de IIS (Internet Information Services), un servidor web diseñado para el sistema operativo Microsoft Windows. Algo más tarde, en 1999 fue lanzado JSP por Sun Microsystems.

Fueron principalmente estas tres tecnologías destinadas a programar la parte del servidor web las revolucionaron el mundo del desarrollo web y permitieron su despegue. Todas ellas fueron diseñadas para el desarrollo de webs con contenido dinámico y presentaban grandes ventajas frente a CGI:

- Las páginas basadas en CGI eran difíciles de mantener y actualizar ya que las instrucciones HTML se encontraban insertadas en el propio código. Esto cambió gracias al modelo MVC (model-view-controller) que separa de forma lógica estos conceptos.
- El rendimiento de CGI en cuanto a páginas dinámicas era muy pobre, ya que cada vez que existía una petición, el servidor tenía que preparar el entorno para ejecutarla, a continuación ejecutarla, devolver su respuesta y destruir el entorno.

Hoy en día estos tres lenguajes siguen siendo muy utilizados (Figura 10), pero se está comenzando a hablar de una nueva generación en el desarrollo web (Figura 11), en la cual se

permite construir aplicaciones web más rápidas y con menos código. Esta nueva generación está basada en la creación de Frameworks web a partir de lenguajes de alto nivel.

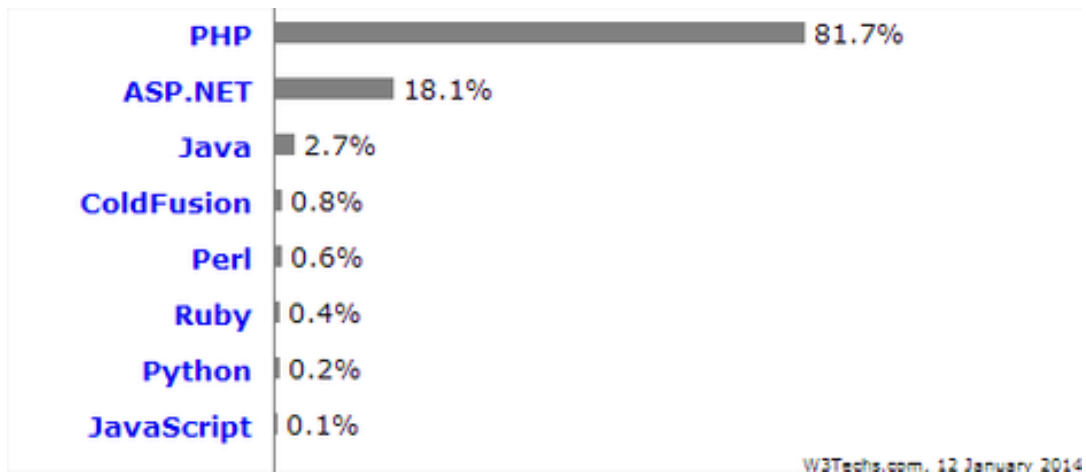


Figura 10: Ratio de los lenguajes del lado del servidor utilizados por sitios web - W3Techs.com

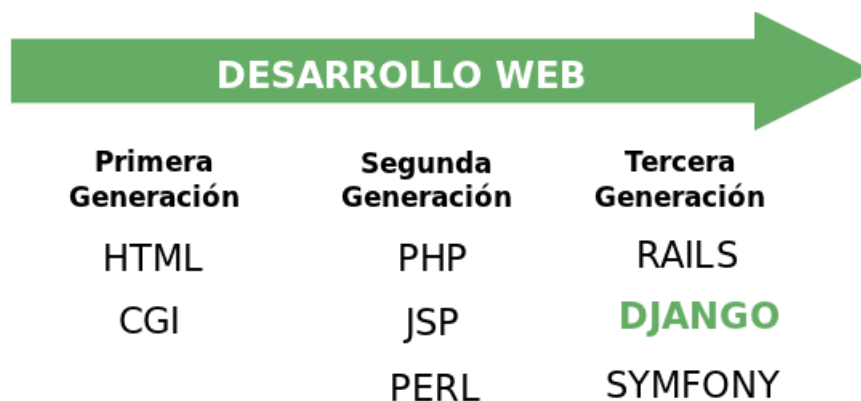


Figura 11: Evolución de las tecnologías web del lado del servidor

Todas ellas (Rails (Ruby), Django (Python), Symfony (PHP)) ofrecen características parecidas y su función es la misma, facilitar el diseño de aplicaciones web.

3.4 DJANGO

Django es una de las frameworks web más populares escrita en Python. Fue inicialmente desarrollada para gestionar aplicaciones web de páginas orientadas a noticias de World Online y más tarde se liberó bajo licencia BSD.

Django impulsa el desarrollo de código limpio al promover buenas prácticas de desarrollo web (como se verá más adelante) y sigue el principio DRY (Don't Repeat Yourself), que fomenta la creación de código más corto y menos repetitivo.

3.4.1 ESQUEMA MTV

Django utiliza una modificación de la arquitectura MVC llamada MTV (model-template-view) con la intención de hacer el framework lo más funcional posible.

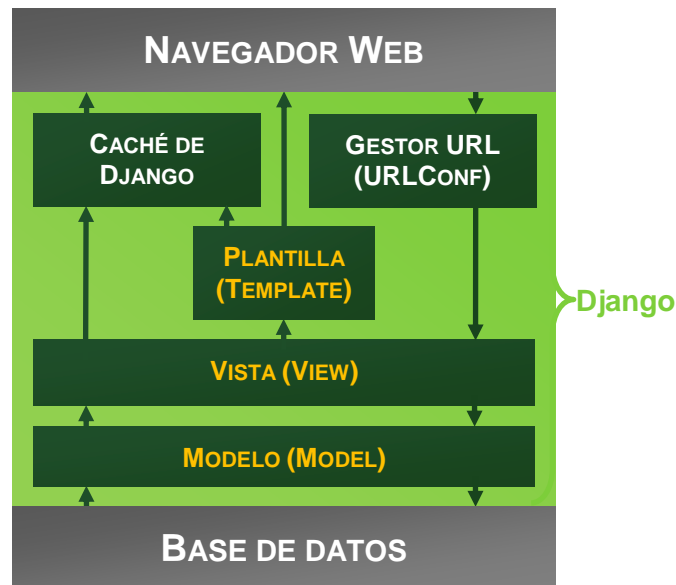


Figura 12: Arquitectura MVC

El esquema de su funcionamiento puede verse mejor en la Figura 12 y una descripción breve del mismo sería:

- El *navegador web* realiza una petición al servidor.
- El gestor de URLs, también llamado *URLConf*, da significado a esta solicitud y ejecuta la función pertinente dentro de la *Vista* apropiada.
- La *vista* en caso de ser necesario, realizará una consulta a la *base de datos* a través del *modelo* y retornará la *plantilla* junto a su contexto (archivos .css, imágenes, etc.) al *navegador web* que se encargará de interpretarlo.

Django cuenta también con una caché de datos donde almacena por ejemplo las claves de las sesiones o los archivos que están descargándose.

Se presta especial atención a las 3 partes principales del esquema:

- **Modelo:** Permite la definición en formas de clases de Python de los datos que serán almacenados. Con esta definición, Django es capaz de crear y/o gestionar una base de datos.

- **Vista:** Controla la respuesta que va a ofrecer el servidor. Puede escribirse tanto en forma de funciones como de clases de Python. Además Django cuenta con un ORM (Object-relational mapping) que permite realizar las consultas a la base de datos mediante código Python gracias al *modelo* definido. También se pueden programar otras tareas conocidas como el envío de correo electrónico o la autenticación y validación de datos a través de formularios.
- **Plantilla:** Podría definirse como la forma en que van a ser presentados los datos. Se encarga de recibir los datos de la vista y organizarlos para la presentación en el navegador. Aquí se utilizan lenguajes de la parte del usuario (HTML, CSS, JS, CSV, etc.), aunque si fuese necesario, también podrían utilizarse etiquetas de Django.

3.4.2 UN PROYECTO EN DJANGO

Django cuenta con comando para la generación de proyectos y aplicaciones automáticas, lo que favorece el orden en el que se presentarán los distintos ficheros del código.

Cuando se crea un proyecto, se generan 5 archivos por defecto:

- **__init__.py:** éste es un archivo vacío usado en python para definir este directorio como un paquete de Python.
- **settings.py:** contiene todas las configuraciones para el proyecto que servirán, por ejemplo, para establecer las rutas de los archivos estáticos o el tiempo para cerrar una sesión.
- **wsgi.py:** contiene una aplicación que permite poner la página en producción junto al servidor web apache y el mod_wsgi. A grandes rasgos, se encarga de indicarle el módulo de Django qué debe utilizar y dónde se encuentra el archivo de configuración del proyecto.
- **urls.py:** en este archivo quedan contenidas las posibles URLs del proyecto; esta información puede estar descentralizada, existiendo un archivo urls.py por cada aplicación. Usa expresiones regulares para su formación.
- **manage.py:** es un script escrito en Python que permite interactuar con nuestro proyecto.

Estos ficheros constituyen la base de cualquier proyecto con la que puede crearse cualquier tipo de web; sólo deben añadirse las aplicaciones que sean necesarias, por ejemplo, una aplicación que se encargue de hacer encuestas, otra aplicación para la exposición de artículos en una tienda online o una aplicación encargada de registro de los usuarios.

4 DESARROLLO DE LA APLICACIÓN

En este capítulo se verán reflejados gran parte de los detalles de programación. Cabe destacar que sólo serán expuestas las decisiones y diseños finales, ya que sería imposible incluir todos los detalles técnicos, errores cometidos o mejoras, que además serían de poca relevancia. El punto comienza haciendo una breve mención a los medios elegidos para el desarrollo, continúa explicando la estructura del proyecto y su configuración, le sigue el diseño del mismo y su porqué y finalmente se describen cada una de las diferentes aplicaciones creadas.

4.1 TOMA DE DECISIONES

En el diseño web, y cualquier campo de la programación, existe una gran diversidad de posibilidades, lo cual puede ser abrumador para un programador novel. Para acotar un poco este área se trató de elegir los medios a utilizar a priori.

El navegador en el que se va a desarrollar la web es **Chrome 36**, lo que significa que no se probará la compatibilidad en otros navegadores. Las razones por las que se ha tomado esta decisión son: está entre los más utilizados (Figura 8), tiene una de las mejores puntuaciones en cuanto a compatibilidad y rendimiento (Figura 9) y además tiene soporte multiplataforma, lo cual lo hace disponible en cualquiera de los sistemas operativos más extendidos (Mac, Windows y GNU/Linux).

Los medios utilizados del lado del servidor se resumen en un sólo lenguaje **Python** (65) (66), en su versión 2.7.5, mientras que la framework web empleada es **Django 1.4.8** (67) (68) (69).

La única decisión tomada antes de comenzar el proyecto fue que las funciones explicadas en el apartado 2 iban a ser implementadas en Python, ya que es un lenguaje de alto nivel y con bibliotecas como Numpy ó Scipy que hacen que su uso resultase muy similar al de la programación en Matlab.

A partir de aquí se buscó una framework web que estuviese escrita en Python, decidiéndose finalmente tomar Django por ser una de las más populares y con mayor actividad. La versión fue elegida en función de la compatibilidad con el servidor. A continuación, se enumeran las bibliotecas (y sus versiones más relevantes) que han sido utilizadas en Python para la implementación de los diferentes algoritmos:

- Numpy 1.4.1
- Scipy 0.12.1
- Sympy 0.7.5

El lenguaje del lado del cliente fue elegido también teniendo en cuenta su popularidad y su potencia, apostando por la combinación de **HTML5**, **CSS3** y **JavaScript** (70) (71), reforzando el mismo con algunas bibliotecas para JS:

- **CanvasJS**: debido a la necesidad de representar una gran cantidad de datos actigráficos, tuvo que elegirse una biblioteca que ayudara a representar los mismos. Los grandes resultados obtenidos en cuanto a velocidad y rendimiento hicieron que se seleccionara ésta en su versión v1.3.0 GA.
- **JQuery**: al tratarse de una página dinámica es necesario modificar los contenidos estáticos sin que esto suponga una carga para el servidor; con la intención de facilitar esta tarea se ha utilizado JQuery 1.11.0.

4.2 ESTRUCTURA DEL PROYECTO

Este punto explica las partes de las que se compone el proyecto, cuya aplicación resultante se ha denominado Actigraphy LPI. La Figura 13 ha sido creada como esquema que sirve de referencia para el resto de apartados. A ella se suma la Figura 14, que es el conjunto de directorios reales que forman el proyecto al completo. Ambas están relacionadas directamente, por lo que serán comentadas simultáneamente.

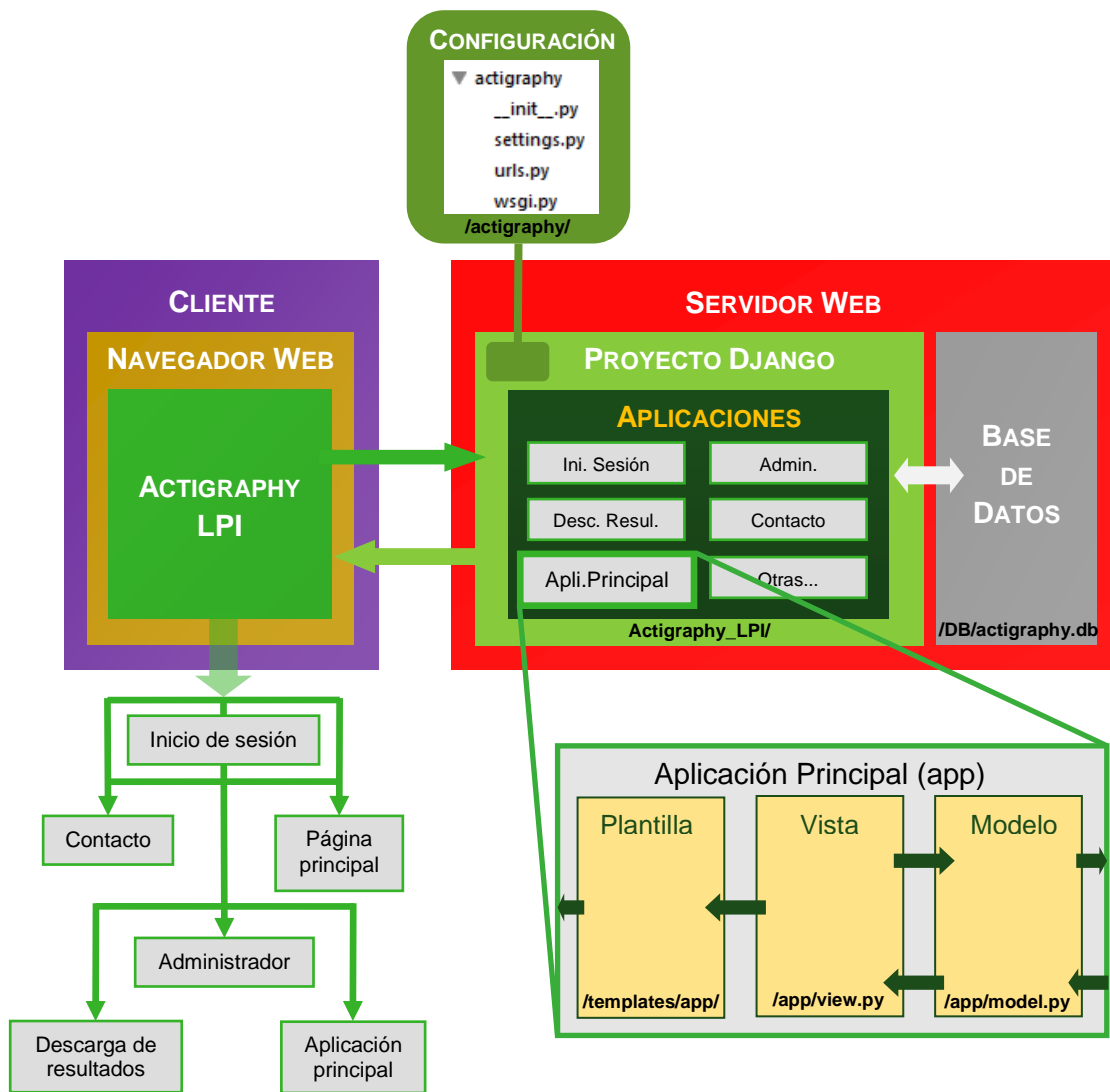


Figura 13: Esquema general de Actigraphy LPI

- ▼ Actigraphy_LPI
 - ▶ actigraphy
 - ▶ app
 - ▶ contact
 - ▶ DB
 - ▶ download_results
 - ▶ log_in_out
 - ▶ other
 - ▶ static
 - ▶ templates
 - ▶ tmp
 - manage.py

Figura 14: Árbol principal del proyecto

La Figura 13 tiene dos partes claramente diferenciadas, la del cliente y la del servidor. El lado del cliente será comentado de forma más detallada posteriormente, aunque sí es necesario saber que viene definido por las "plantillas".

En el lado del servidor es donde está ubicada la carpeta "*actigraphy_LPI*" de la Figura 14 que puede dividirse en tres grupos:

- **Configuración:** contiene los archivos "base" del proyecto que se encuentran en la carpeta "*actigraphy*". A su vez está formado por: `__init__.py`, `settings.py`, `urls.py` y `wsgi.py`.
- **Base de Datos:** es el directorio "*DB*" y contiene la base de datos llamada "*actigraphy.db*". En ésta se van a almacenar todos los datos necesarios para Actigraphy LPI. La razón de que se encuentre en un directorio propio es porque el sistema de gestión utilizado (SQLite3) necesita tener permisos de escritura y lectura en el mismo.
- **Aplicaciones:** es el conjunto de herramientas que va a conformar el proyecto, y se fracciona en dos partes lógicas:
 - La del servidor, al que pertenecen los directorios *app*, *contact*, *download_results*, *log_in_out* y *other*, todas ellas formadas a su vez por un subconjunto de los archivos:
 - `__init__.py`: define el directorio como un paquete de Python.
 - `admin.py`: este archivo especifica a la aplicación de administrador de Django cómo deben ser representados los datos dentro de la misma.
 - `models.py`: es el encargado de definir los datos que serán almacenados, a partir del mismo. Django genera las tablas donde se guardará la información estableciendo también el tipo de información que será almacenada y otras opciones como el valor por defecto.
 - `views.py`: contiene las funciones o clases que definen y controlan las respuestas del servidor a las peticiones del usuario haciendo uso de la información recibida, como puede ser la ofrecida por el localizador URL.
 - La del cliente, compuesta por los directorios:
 - *static*: contiene todos los archivos que sean de carácter estático como scripts de JavaScript, imágenes y otros recursos.
 - *templates*: en esta carpeta se encuentran los archivos HTML de cada una de las aplicaciones. Se organizan en subdirectorios con el mismo nombre que la aplicación a la que pertenecen (Figura 15).

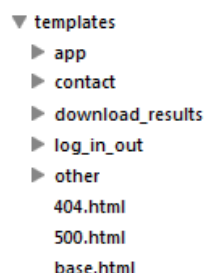


Figura 15: Árbol del directorio templates

Por último, la carpeta "tmp", está destinada a archivos de carácter temporal y requiere de privilegios de lectura y escritura.

4.3 CONFIGURACIÓN DEL PROYECTO

Como se ha explicado anteriormente, al generar un proyecto se crean varios archivos. En este apartado se explican settings.py y urls.py., cuya configuración supone la base de Actigraphy LPI

4.3.1 SETTINGS.PY

El archivo settings.py es el responsable de definir el marco del proyecto.

El primer paso consiste en crear una variable que contenga la ruta absoluta del proyecto; como ésta puede cambiar entre la fase de desarrollo y la fase de producción, se crea de forma paramétrica. La variable BASE_DIR pasará a contener ahora la ruta al directorio 'root' actigraphy.

```
import os
BASE_DIR = os.path.dirname(os.path.dirname(__file__))
```

Debugger

La variable DEBUG permite activar o desactivar el debugger de Django. Éste pasará a estar desactivado cuando la aplicación esté en modo producción.

```
DEBUG = True #Debugger activado
DEBUG = False #Debugger desactivado
```

Definir el administrador

La definición del administrador puede hacerse utilizando una tupla asignada a la variable ADMINS. La forma correcta en la que debe darse esta información sigue la estructura ('nombre', 'email').

```
ADMINS = (
    ('Daniel T', 'actigrafia@gmail.com'),
)
```

Definir la base de datos y el directorio

La base de datos se define utilizando un diccionario de python. En este caso la base de datos será gestionada usando SQLite3. Para su configuración en este archivo deben definirse dos datos: ENGINE que indica a Django dónde está el paquete con el que gestionar esta base de datos y la definición NAME da la ruta donde se encuentra la base de datos.

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.sqlite3',
        'NAME': BASE_DIR + '/DB/actigraphy.db',
    }
}
```

Definir host.

En este parámetro puede definirse el dominio de la web. Para permitir un dominio cualquiera, puesto que a priori éste era desconocido, se utiliza la siguiente lista de Python.

```
ALLOWED_HOSTS = ['*']
```

Definir la zona horaria

Django permite seleccionar la zona horaria (72).

```
TIME_ZONE = 'Europe/Madrid'
```

Definir los directorios

Es posible definir directorios para poder emplearlos posteriormente con diferentes fines. STATIC_ROOT¹ se utiliza para archivos de carácter estático (como los CSS, las imágenes o los scripts de JS). Finalmente se define el directorio para las plantillas TEMPLATE_DIR, donde estarán ubicadas todas las plantillas que ejecutarán las vistas.

```
if DEBUG:
    STATIC_ROOT = os.path.join(BASE_DIR, '/static')
else:
    STATIC_ROOT = os.path.join(BASE_DIR, 'static')
TEMPLATE_DIRS = (
    os.path.join(BASE_DIR, 'templates'),
)
```

¹ Está incrustado dentro de un bucle if debido a un bug de Django en esta versión, que toma la ruta de una forma u otra en función de si el modo depurador está activado.

Definir URL

Para poder hacer uso de los archivos estáticos en el entorno web, es necesario definir cómo será la URL de forma relativa.

```
STATIC_URL = '/static/'
```

Definir archivo URL principal

Esta variable debe hacer referencia al archivo "urls.py" principal. Para respetar el orden y los nombres establecidos por Django, se define éste como ROOT_URLCONF.

```
ROOT_URLCONF = 'actigraphy.urls'
```

Configuración de la sesión

Al acceder a una página web, ésta crea una sesión para el usuario. Mediante algunas variables podemos controlar parte de este funcionamiento, en concreto, si la sesión es cerrada cuando el navegador se cierra:

```
SESSION_EXPIRE_AT_BROWSER_CLOSE=True
```

Configuración de upload files

También es posible configurar desde aquí el tamaño máximo de los archivos que pueden ser subidos (FILE_UPLOAD_MAX_MEMORY_SIZE) así como el destino temporal de los mismos (FILE_UPLOAD_TEMP_DIR).

```
FILE_UPLOAD_MAX_MEMORY_SIZE = 31457280 #30MB  
FILE_UPLOAD_TEMP_DIR = os.path.join(BASE_DIR, 'tmp')
```

Configuración del Email

Se configura el sistema para que los correos, tanto para el contacto como para el envío de errores sean funcionales.

```
#Email  
#name-> Daniel Actigrafia  
EMAIL_USE_TLS = True  
EMAIL_HOST = 'smtp.gmail.com'  
EMAIL_HOST_USER = 'actigrafia@gmail.com'  
EMAIL_HOST_PASSWORD = 'ActigrafiaDaniel2014'  
EMAIL_PORT = 587
```

Definición de las aplicaciones

También es necesario indicar las aplicaciones que pasarán a formar parte del proyecto principal; éstas pueden ser propias o procedentes de Django. Las aplicaciones utilizadas provenientes de Django están situadas al principio y sirven para el envío de correos, el uso de archivos estáticos, la administración de sesión y de usuario o para definir el tipo de contenidos que serán representados de cara al usuario.

```
INSTALLED_APPS = (  
    'django.contrib.auth',  
    'django.contrib.contenttypes',  
    'django.contrib.sessions',  
    'django.contrib.sites',  
    'django.contrib.messages',  
    'django.contrib.staticfiles',  
    'django.contrib.admin',  
    # Own  
    'app', #Application  
    'download_results', #Download Results  
    'contact', #Contact  
    'log_in_out', #Log-in/Log-out  
    'other', #Main page and other  
)
```

4.3.2 URLS.PY

Este archivo define cuáles son las funciones que ejecutará el servidor en función de la URL.

Su código es de gran simplicidad, y se lleva a cabo utilizando una función que toma como argumentos una expresión regular, encargada de definir la URL, y una ruta, destinada a indicar la función que debe ejecutarse. Se presenta a continuación el código para cada una de las aplicaciones.

Principal:

```
#Main  
url(r'^$', 'other.views.main'),
```

App:

```
#App
url(r'^app/$', 'app.views.app'),
url(r'^register_graphic/$', 'app.views.register_graphic'),
#Load file (choose file, delete file, add file)
url(r'^select_file/$', 'app.views.select_file'),
url(r'^delete_file/$', 'app.views.delete_file'),
url(r'^upload_file/$', 'app.views.upload_file'),
url(r'^process_data/$', 'app.views.process_data'),
#Show info
url(r'^show_info/$', 'app.views.show_info'),
url(r'^edit_info/$', 'app.views.edit_info'),
#General
url(r'^set_general/$', 'app.views.set_general'),
#Statistical analysis
url(r'^stat_graphic/$', 'app.views.stat_graphic'),
#Sleep analysis
url(r'^sleep_graphic/$', 'app.views.sleep_graphic'),
#Cosinor analysis
url(r'^cosinor_graphic/$', 'app.views.cosinor_graphic'),
#Symbolic analysis
url(r'^symbol_graphic/$', 'app.views.symbol_graphic'),
#ctm
url(r'^ctm_graphic/$', 'app.views.ctm_graphic'),
#Classifier
url(r'^classifier_graphic/$', 'app.views.classifier_graphic'),
```

Download results

```
#Download Results
url(r'^download_results/$', 'download_results.views.download_results'),
url(r'^generate_results/$', 'download_results.views.generate_results'),
url(r'^get_results/$', 'download_results.views.get_results'),
```

Contact

```
#Contact
url(r'^contact/$', 'contact.views.contact'),
```

Archivos estáticos y otros

```
url(r'^static/(?P<path>.*?)$', 'django.views.static.serve', {'document_root':
settings.STATIC_ROOT, 'show_indexes': settings.DEBUG}),
url(r'^loading_1/$', 'other.views.loading_1'),
url(r'^loading_2/$', 'other.views.loading_2'),
url(r'^uploading_file/$', 'other.views.uploading_file'),
url(r'^processing_data/$', 'other.views.processing_data'),
```

Log-in / Log-out

```
#log
url(r'^log_in/$', 'log_in_out.views.to_log_in'),
url(r'^log_out/$', 'log_in_out.views.to_log_out'),
```

Admin

```
#Admin
url(r'^admin/', include(admin.site.urls)),
```

4.4 DISEÑO Y ARQUITECTURA DE LA PÁGINA WEB

Dentro de la programación web existe una parte de diseño gráfico que tiene como fin crear una interfaz de usuario. Ésta abarca todo lo que el cliente ve cada vez que visita la página. Para que la experiencia sea lo más agradable posible es necesario fijar dos ideas: la arquitectura y el diseño de la página.

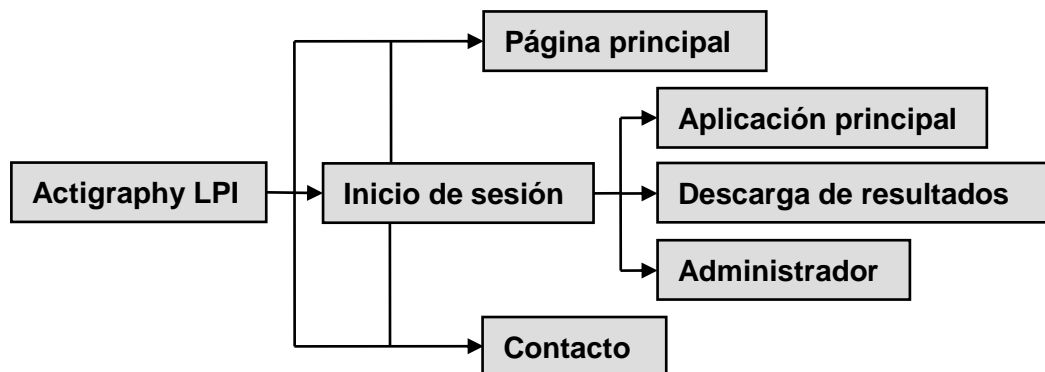


Figura 16: Arquitectura de Actigraphy LPI

La arquitectura básica de la página puede verse en la Figura 16, en la que se observan 6 aplicaciones:

- **Página principal:** es la página de presentación para la web.
- **Inicio de sesión (Log-in/Log-out):** es la aplicación mediante la cual puede restringirse el acceso de los usuarios a ciertas partes de la página.

- **Aplicación principal (Application):** es la aplicación que permite estudiar el registro actigráfico detalladamente.
- **Descarga de resultados (Download results):** es la aplicación que permite generar una hoja de cálculo con descriptores numéricos del registro.
- **Administrador (Admin):** es la aplicación mediante la cual el usuario de mayor rango podrá administrar la página web.
- **Contacto (Contact):** es la aplicación que permite la comunicación de los usuarios con los creadores de Actigraphy LPI

Para el diseño de la web se tuvieron en cuenta tres aspectos:

- *Los usuarios:* Podría definirse a los mismos como personas adultas que poseen grandes conocimientos en sus respectivos áreas, tanto medicina como psicología.
- *El objetivo de uso:* La aplicación va a ser utilizada para trabajar, no tiene un ánimo ocioso.
- *La plataforma:* Van a ser ordenadores de escritorio y portátiles con pantallas sin grandes resoluciones.

Partiendo de estos datos, se trata de empatizar con el usuario y se establecen objetivos a raíz de los siguientes razonamientos:

- La aplicación debe ser intuitiva para que los usuarios no tengan que dedicar tiempo al aprendizaje de un nuevo software.
- La aplicación puede ser utilizada durante bastante tiempo, por lo que la composición de colores debe ser relajante para que no canse la vista. En cuanto a los objetos mostrados simultáneamente, su número debe ser limitado, de forma que se encuentre fácilmente lo que se busca.
- Los botones y las letras deben tener un tamaño adaptativo ya que la resolución de las pantallas puede variar y esto no debe afectar a la arquitectura visual de la página. La aplicación ha de aprovechar la pantalla de forma eficiente para mostrar los resultados
- Para finalizar, la aplicación debe ser lo más rápida posible; en particular, el trasiego de datos entre cliente y servidor debe ser limitado al mínimo imprescindible.

Estas pautas serán tenidas en cuenta a lo largo de todo el diseño gráfico de la aplicación.

Llegado este punto, ya es posible hablar del diseño base de Actigraphy LPI que servirá de soporte para las aplicaciones. Se ha elegido una estructura clásica de cabecera y pie de página; ambos se sitúan de forma fija en los extremos superior e inferior del navegador, a no ser que:

- La ventana del navegador sea demasiado pequeña, en cuyo caso aparecerá un scroll. El tamaño mínimo en horizontal es de 1100px (fijado por la cabecera) y en vertical de 846px (fijado por el contenido).
- Que el contenido que se desea mostrar sea excesivamente grande; en tal caso, se recurrirá a colocarlo verticalmente para que el contenido visible en pantalla sea asequible y pueda seguirse con facilidad. Esto sucede, por ejemplo, en la aplicación "download results"

Además, esta configuración sirve de guía al usuario, que rápidamente puede identificar cuándo empieza y finaliza la información que se le desea mostrar. También se utiliza la cabecera como menú de aplicaciones, para que el usuario disponga de ellas rápidamente.

La composición de colores está constituida por una gama de grises y verdes con tonalidades apagadas.

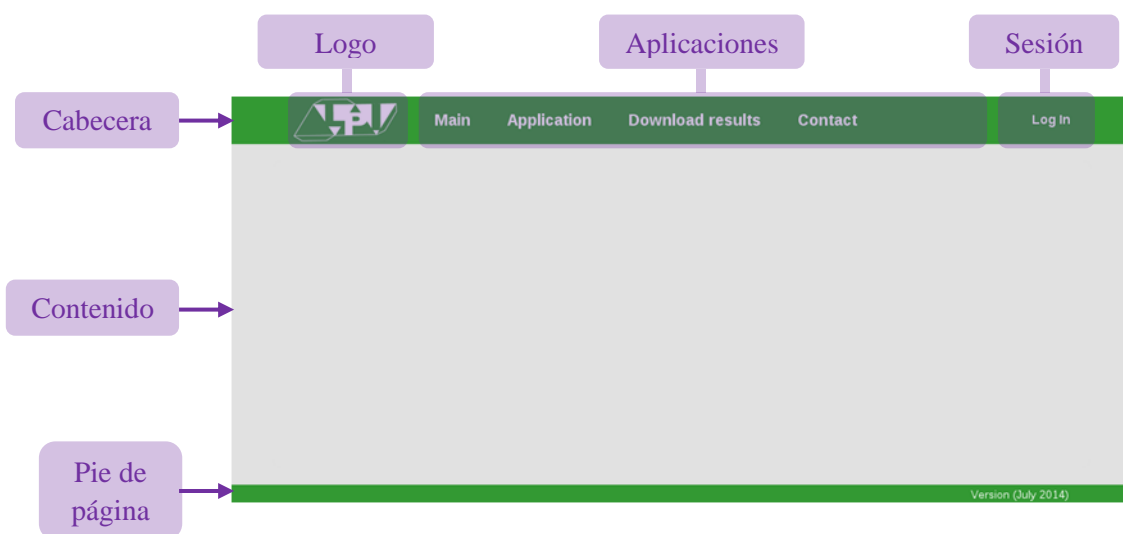


Figura 17: Esquema base de la página web

La Figura 17 muestra el resultado final. El sistema de plantillas de Django hace que podamos establecer ésta como base del resto de páginas webs. A este archivo se le ha denominado "base.html" y se exportará siempre al comienzo de la página principal de cada aplicación. A modo de observación, queda decir que el fondo del contenido no es blanco si no gris, ya que un gran fondo blanco en entornos con poca luz puede resultar molesto y ser muy cansado para la vista. Otro aspecto que puede llamar la atención es que la aplicación de administrador no está disponible, permaneciendo oculta a la mayoría de los usuarios, ya que nadie excepto el administrador debe utilizar esta herramienta.

Establecidas ya las bases y el diseño, es momento de centrarse en las aplicaciones. Antes de comenzar, se explicará la metodología que va a ser seguida para describir su diseño y funciones.

Cada aplicación está formada por todas o algunas de las siguientes partes:

- *Modelo (Model)*: siguiendo el nombre que propone Django, se le llamará modelo al diseño de las tablas, sus relaciones y las características de los datos que se encontrarán en la base de datos. Para su detalle se expondrá el código de Python y se realizará la explicación a partir del mismo.
- *Diseño de la interfaz y peticiones al servidor*: en este apartado no se analizará el código HTML, sino que se esquematizará la parte gráfica de la aplicación y se enumerarán las características que posee, cómo se hace la petición al servidor en cada caso y por qué.
- *Vista (Views)*: Se describirán los algoritmos seguidos por las funciones para ofrecer una respuesta a la petición de los usuarios.

4.5 APPLICATION

Es la aplicación principal y la más compleja. Su fin es poder analizar la información actigráfica detenidamente mediante el uso de los algoritmos explicados anteriormente, pero también es el lugar donde se gestionan los registros actigráficos. La Figura 18 muestra el árbol que forma el código de la parte del servidor, todo lo que depende de la carpeta "new_program" es referente a la implementación de los algoritmos descritos en el punto 2.

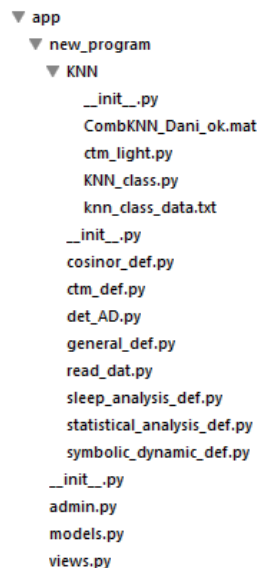


Figura 18: árbol de la "application"

4.5.1 MODELO

Las tablas que conforman la base de datos han sido definidas mediante clases en Python y se distinguen dos grupos, las destinadas a contener la información de los registros (Figura 19) y las destinadas a contener la información de la sesión actual de cada usuario (Figura 20).

Los datos de los registros son cargados directamente del archivo generado por el actígrafo, a los mismos se les añaden otros como la máscara de actividad-descanso o el módulo de la suma de los vectores x, y, z.

```
#####
# FILE DATA #
#####
class file_data(models.Model):
    user_name = models.ForeignKey(User)
    file_name = models.CharField(max_length=200)
    upload_date = models.DateTimeField('upload_date', auto_now=True)
    start_time = models.CharField(max_length=100, blank = True, null=True)
    start_date = models.CharField(max_length=100, blank = True, null=True)
    epoch_period = models.CharField(max_length=100, blank = True, null=True)
    mode = models.IntegerField(blank = True)
    descriptive_name = models.CharField(max_length=200)
    notes = models.CharField(max_length=1000, blank = True, null=True)
    n_channels = models.IntegerField(blank = True, null=True)
    channels_name = models.CharField(max_length=100, blank = True, null=True)
    total_t = models.CharField(max_length=100, blank = True, null=True)
    data_per_channel = models.IntegerField(blank = True)
    a_r_start = models.CharField(max_length=100, blank = True, null=True)
    a_r_end = models.CharField(max_length=100, blank = True, null=True)
    interval_end = models.CharField(max_length=100, blank = True, null=True)
    def __unicode__(self):
        return self.descriptive_name

class file_header(models.Model):
    regarding = models.ForeignKey(file_data, unique=True)
    header = models.TextField(blank = True, null=True)
    def __unicode__(self):
        return unicode(self.regarding)

class file_channel(models.Model):
    regarding = models.ForeignKey(file_data)
    number = models.IntegerField(blank = True)
    channel = models.TextField(blank = True, null=True)
    def __unicode__(self):
        return unicode(self.regarding)

class file_mask(models.Model):
    regarding = models.ForeignKey(file_data, unique=True)
    mask = models.TextField(blank = True, null=True)
    def __unicode__(self):
        return unicode(self.regarding)
```

Figura 19: Modelo para los registros actigráficos

La tabla maestra es "file_data" y a su vez está relacionada con la del usuario. En ella se almacenan los datos de menor tamaño que pueden ser accedidos de forma habitual cuando se trabaje con la aplicación. Existen otras 3 tablas que están relacionadas con la maestra y que también contienen datos del registro original:

- file_header: es la que contiene la cabecera del archivo original, que se conserva para poder reproducir rápidamente el archivo original si fuese necesario.
- file_channel: contiene la información de cada uno de los canales por separado.
- file_mask: contiene la máscara de actividad-descanso calculada al subir el registro.

A modo de referencia se indica el significado de las variables almacenadas:

- file_data
 - user_name: el usuario al que pertenece.
 - file_name: nombre del archivo original.
 - upload_date: fecha en la que se ha subido.
 - start_time: hora, minuto y segundo en los que comenzó el registro.
 - start_date: año, mes y día en los que comenzó el registro.
 - epoch_period: periodo de muestreo (hh:mm:ss).
 - mode: modo de configuración del actígrafo.
 - descriptive_name: nombre descriptivo asignado por el usuario para poder identificar al registro en el futuro.
 - notes: notas escritas por el usuario referentes al registro.
 - n_channels: el número total de canales del registro.
 - channels_name: el nombre de cada uno de los canales.
 - total_t: tiempo total del registro.
 - data_per_channel: longitud del cada canal.
 - a_r_start: índices del canal donde comienza cada periodo de descanso.
 - a_r_end: índices del canal donde termina cada periodo de descanso.
 - interval_end: momento en el que termina el registro (dd:hh:mm:ss).
- file_header
 - regarding: referencia a la tabla maestra.
 - header: texto de la cabecera sin modificar.
- file_channel
 - regarding: referencia a la tabla maestra.
 - number: número del canal.
 - channel: datos del canal.
- file_mask
 - regarding: referencia a la tabla maestra.
 - mask: máscara de actividad descanso (formada por "unos" y "ceros").

```
#####  
#     SESSION DATA     #  
#####  
class session_data(models.Model):  
    user_name     = models.ForeignKey(User,unique=True)  
    session_key   = models.TextField()  
    file_dat      = models.ForeignKey(file_data,  
        blank     = True,  
        null      = True,  
        on_delete = models.SET_NULL)  
    channel       = models.IntegerField(blank = True,default=0)  
    channel_name  = models.CharField(max_length=200,blank = True, null=True)  
    decim         = models.IntegerField(blank = True,default=1)  
    onset         = models.CharField(max_length=100,blank = True, null=True)  
    end           = models.CharField(max_length=100,blank = True, null=True)  
    def __unicode__(self):  
        return unicode(self.user_name)  
  
class session_channel_mask(models.Model):  
    regarding     = models.ForeignKey(session_data, unique=True)  
    channel       = models.TextField(blank = True, null=True)  
    mask          = models.TextField(blank = True, null=True)  
    def __unicode__(self):  
        return unicode(self.regarding)
```

Figura 20: Modelo para la información de sesión

Los datos de sesión son utilizados para proporcionar a los algoritmos información del registro modificada, que se almacena para evitar que tenga que ser recalculada cada vez, también sirve para que el usuario pueda retomar su trabajo en el mismo punto que lo dejó.

La tabla principal es "session_data" y está a su vez referenciada con la tabla del registro original y la tabla del usuario. La tabla "session_channel_mask" contiene los datos del registro actual con las modificaciones sufridas y de la máscara de actividad-descanso.

- session_data
 - user_name: referencia al usuario.
 - session_key: contiene la clave de la sesión actual (su significado se verá más adelante).
 - file_dat: referencia a la tabla del registro original.
 - channel: es el número que corresponde al canal actual.
 - channel_name: el nombre del canal actual.
 - decim: el diezmado que se le está aplicando al canal.
 - onset: comienzo del canal actual.
 - end: fin del canal actual.
- session_channel_mask
 - regarding: referencia a la tabla session_data.
 - channel: datos del canal actual.
 - mask: datos de la máscara de actividad descanso actual.

Todos los datos cuyo campo no sea ForeignKey o IntegerField son guardados como cadenas de caracteres; en el caso de tratarse de vectores, cada posición está separada por una coma. Al leer esta información de la base de datos, se transforma al tipo de dato adecuado para su manipulación en Python.

4.5.2 DISEÑO DE LA INTERFAZ Y PETICIONES AL SERVIDOR

Esta gran aplicación se divide en un conjunto de herramientas que serán denominadas como:

- **Cargar archivo (Load File):** es la parte donde el usuario puede subir un nuevo registro y seleccionar o eliminar uno ya existente en la base de datos.
- **General:** Permite modificar el registro que será utilizado para el cálculo de los algoritmos.
- **Análisis estadístico (Statistical Analysis)**
- **Análisis de sueño (Sleep Analysis)**
- **Cosinor (Cosinor Analysis)**
- **Dinámica simbólica (Symbolic Dynamics)**
- **CTM**
- **Clasificador (Classifier)**

El diseño es similar en cada una de ellas excepto en "Cargar archivo", que varía un poco debido a sus características. Toda la aplicación es una única página web, la cual recibe los datos del servidor y los muestra al usuario.

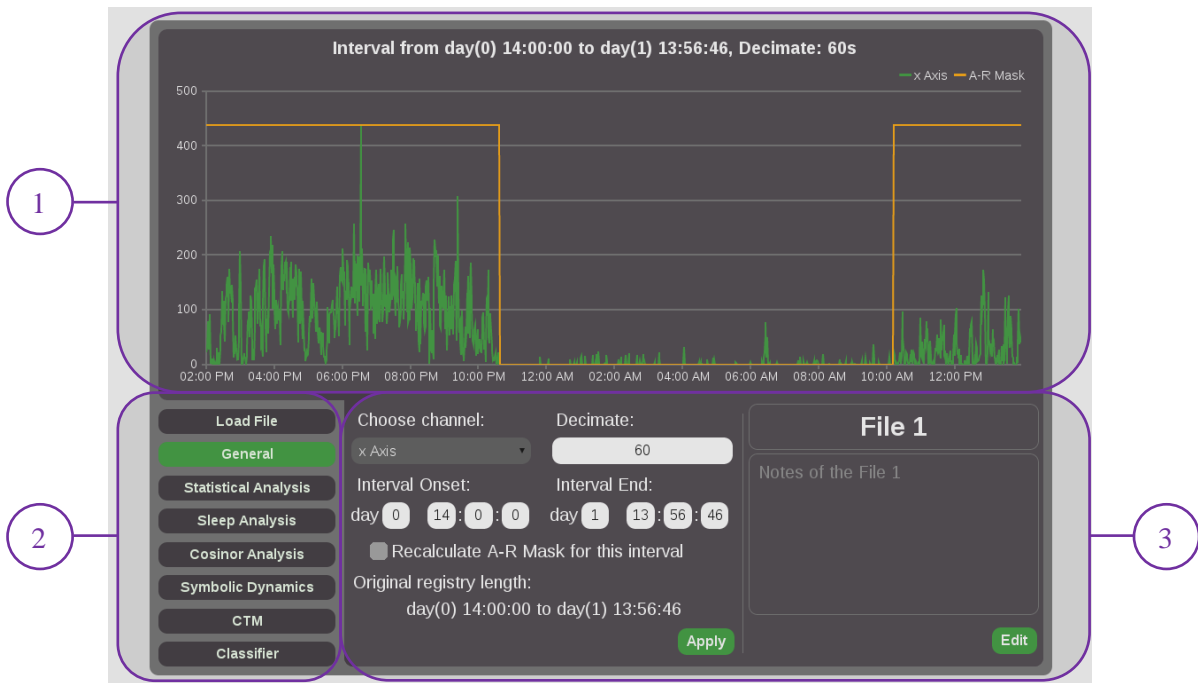


Figura 21: Diseño general de la aplicación

El diseño general de la aplicación se muestra en la Figura 21 y puede dividirse en tres grandes áreas:

1. **Zona de resultados** : en esta zona se mostrarán tanto los resultados numéricos como las gráficas obtenidas. Se ha elegido la parte superior de la pantalla porque es la más amplia. Su tamaño variará tanto en horizontal como en vertical en función de las dimensiones de la pantalla y al representar gráficas. Debido a su carácter dinámico, está formada por distintos contenedores dependiendo de la actual herramienta. Cada uno de ellos contiene una "iframe" que permite cargar una web dentro de otra; de esta forma sólo se actualiza esa parte de la página, evitando que tenga que reenviarse toda la información.
2. **Zona de botones** : muestra todas las herramientas implementadas en la aplicación mediante una cadena de botones colocados verticalmente. Se ha elegido la zona lateral inferior derecha por ser un lugar donde habitualmente se sitúan botones (como por ejemplo el botón de Windows). Sólo con pulsar en uno de estos botones, cambiará tanto la zona de resultados como la zona de formularios.
3. **Zona de formularios** : mostrará las diferentes posibilidades que nos permite cada una de las herramientas, ya sea para el cálculo de características, la administración de los archivos o la toma de notas. Está formada por contenedores dentro de la misma página web, salvo la zona de la toma de notas, que en su interior contiene una iframe

para que pueda guardarse el contenido enviado sin la necesidad de actualizar toda la web.

Este patrón se repetirá en todas las partes de la aplicación para dar coherencia al conjunto, continuando con la combinación de colores planos y apagados mencionada en puntos anteriores. Para comprender algo mejor este esquema, la Figura 22 muestra el archivo "app.html" donde se encuentra escrita la página principal, las subpáginas que son cargadas las que penden de la carpeta "graphics" (se encargan de mostrar los resultados) y la página "show_info.html", que es el apartado donde el usuario puede introducir notas o cambiar el nombre del registro.

```

▼ templates
  ▼ app
    ▼ graphics
      cosinor_graphic.html
      ctm_graphic.html
      knn_graphic.html
      register_graphic.html
      sleep_graphic.html
      stat_graphic.html
      symbol_graphic.html
    app.html
    show_info.html
  
```

Figura 22: Esquema de las plantillas de "application"

El contenido representado en la zona de resultados variará en función de la herramienta, pudiendo encontrar graficas 2D (lineales o de puntos), tablas, histogramas o un diagrama fasorial (Figura 23).

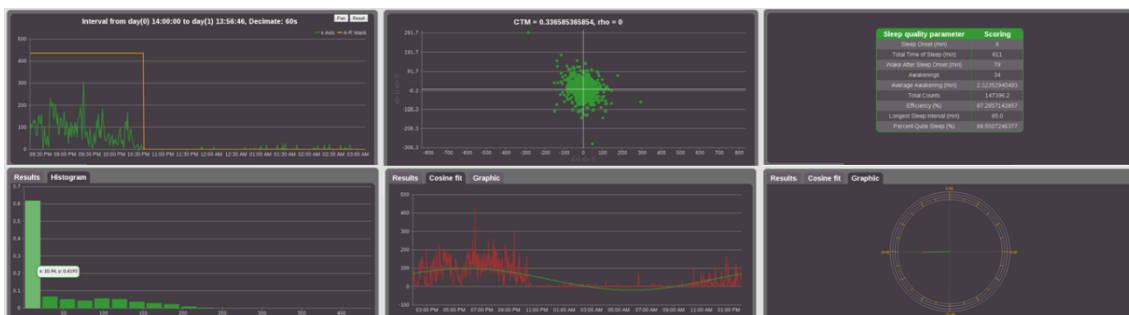


Figura 23: Ejemplo de los diferentes datos mostrados

En el caso el resultado todavía no esté calculado, se mostrará el mensaje pertinente en el lugar de los datos.

Los formularios son también totalmente dinámicos, lo que significa que si algún campo no se cumplimenta adecuadamente se mostrará al usuario la información para la corrección del mismo, proporcionada la primera vez que es enviada la página. Además, los formularios cuentan con valores por defecto que facilitan la labor del usuario.

Las preguntas que se hace la página para la comprobación de los formularios son las siguientes:

- ¿Es un número entero? Comprueba que el número sea un entero.

- *¿Están dentro de la cota inferior o superior?* Comprueba si el número pertenece al intervalo, que puede ser predefinido, estar proporcionado por el servidor o incluso depender de los datos rellenados en el formulario en otros apartados, tal y como sucede en el de la dinámica simbólica. Esta se utiliza también para comprobar si son horas, minutos o segundos junto a la primera cuestión.
- *¿Pertenece al intervalo adecuado de tiempo?* Comprueba si el intervalo que se está pidiendo entra dentro del registro original
- *¿Es el comienzo del intervalo inferior al final del intervalo?* Los intervalos están dados por el formato dd:hh:mm:ss, este se pasa a segundos y se comprueba.
- *¿Tiene la longitud adecuada?* Se usa para entradas de caracteres, aunque hay una variación para comprobar las horas, los minutos y los segundos, ya que éstos no pueden estar formados por más de dos dígitos cada uno.

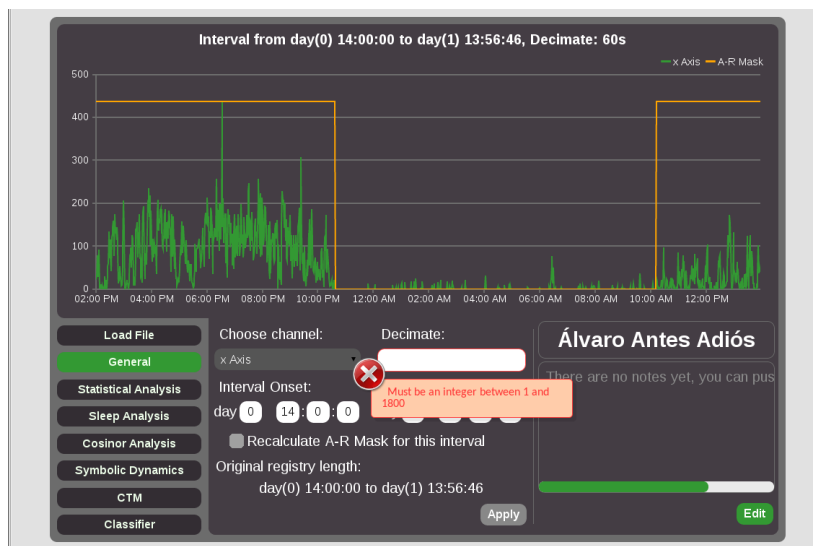


Figura 24: Error en un formulario

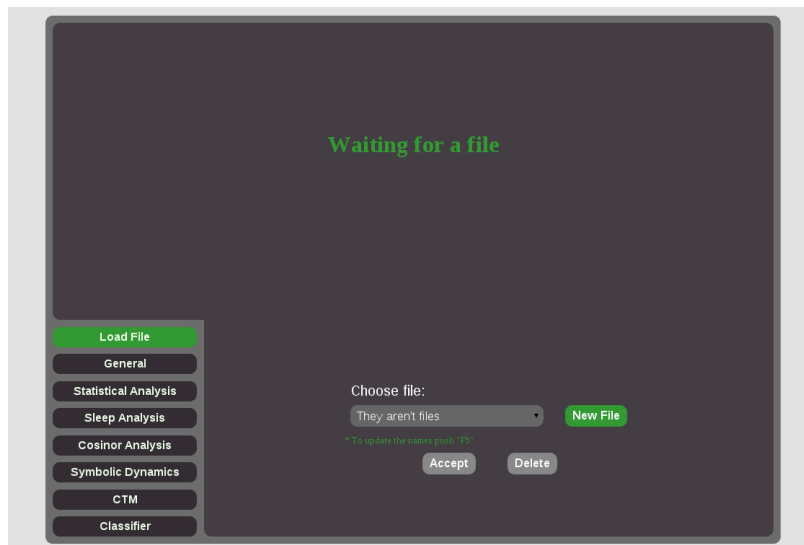


Figura 25: Aplicación sin ningún registro

En caso de que la respuesta a estas preguntas sea errónea, inmediatamente se deshabilitará el botón para enviar el formulario y se mostrará el error (Figura 24). Cuando la respuesta sea correcta, se habilitará de nuevo.

Estas eventualidades son tenidas en cuenta también en el caso de que no exista ningún registro subido (Figura 25), o el archivo que se desea subir no tenga las características deseadas.

En los formularios también se realiza la petición al servidor, ya sea mediante el uso de AJAX ó de JavaScript y se clasificarán en dos grupos:

- **Recargar página completa:** una vez realizado el proceso la página será recargada completamente. Esto sucede cada vez que es necesario actualizar los datos empleados en los formularios, como cuando se sube un nuevo archivo, cuando se selecciona un nuevo registro y cuando se aplican cambios en el registro actual mediante la herramienta "General".
- **Recargar una parte parcial de la página:** sucede en el resto de ocasiones al pulsar los botones implementados en las diferentes herramientas.

4.5.3 VISTA

Está formado por diferentes herramientas y éstas a su vez por diferentes funciones. Todas las funciones requieren estar registrado, o lo que es lo mismo, que el usuario al que pertenezca la actual sesión no sea anónimo.

Página principal

Está definida por la función "app", su algoritmo es:

- Carga la información necesaria de la base de datos.
- Se la envía a la página "app.html" para que ésta pueda completar los formularios.

Load file

Esta función abarca los usos de subir un nuevo registro y seleccionar o eliminar un registro ya existente.

Función "*select_file*"

- Comprueba que ha recibido la ID (identificación) correcta.
- En caso afirmativo carga el registro de la base de datos y actualiza las tablas de la sesión del usuario.
- En caso negativo o de error retorna el resultado que indica a la web que la operación ha fallado.

Función "*delete_file*"

- Comprueba que ha recibido la ID (identificación) correcta.
- En caso afirmativo:
 - Elimina el registro de la base de datos
 - Si el registro elegido es el mismo que el actual, también borra dicha información de la tabla de la sesión del usuario.
- En caso negativo o de error retorna el resultado que indica a la web que la operación ha fallado

Función "*upload_file*"

- Comprueba que el archivo cumple con las siguientes características:
 - Tamaño inferior a 30MB.
 - Extensión ".dat".
 - El nombre del archivo es menor de 40 caracteres.
- En caso afirmativo:
 - Guarda el archivo en la carpeta "tmp/files_dat/" con el nombre del usuario combinado con el nombre del archivo.
 - Guarda la dirección en una variable y se la envía como retorno a la página web, indicando que se ha subido con éxito.
- En caso negativo:
 - Envía a la web una variable que indica el tipo de error sucedido

Función "*process_data*". Esta función se ejecuta después de que el archivo ha sido subido con éxito.

- Lee la dirección donde se ha guardado el archivo, el nombre descriptivo y las notas proporcionadas por el usuario.
- En caso de que los datos sean correctos:
 - Lee el registro.
 - Calcula la máscara de actividad-descanso con el "Eje X" (habitualmente es el canal 0)
 - Guarda todos los datos en la base de datos
 - Actualiza la base de datos de la sesión actual cargando el registro que se acaba de subir con la configuración por defecto.
- En caso de que los datos no sean correctos, se retorna un error a la página.

General

Representa el registro actual y realiza los cambios solicitados por el uso del formulario de la herramienta general. En esta herramienta también se abarca la función que permite modificar el nombre y las notas relativas al registro.

Función "*show_info*" crea la gráfica del registro.

- Lee el canal y la máscara contenidos en la tabla de la sesión y se los envía a la web "show_info.html" para su representación.
- Si estos datos no existen, envía una variable a la web para indicárselo.

Función "*edit_info*".

- Comprueba que se han recibido las variables que contienen el nombre descriptivo ("name") y las notas ("note").
- Sobrescribe estas variables en la base de datos.
- En caso de que este proceso falle se le retornará un error a la página.

Función "*set_general*".

- Lee los datos enviados por el formulario de la herramienta "general".
- Carga el registro original actual de la base de datos con el canal especificado.
- Realiza la acotación del intervalo.
- Calcula la máscara de actividad-sueño si es necesario.
- Realiza el diezmado del canal y del registro.
- Actualiza la tabla de sesión de la base de datos.
- Envía a la web una respuesta que el indica que todas las operaciones se han realizado correctamente.

Statistical Analysis

Realiza el análisis estadístico.

Función "*stat_graphic*".

- Comprueba los datos que le han llegado.
- Si no ha llegado ningún dato:
 - Genera una variable "var" con valor 2.
- Si se han recibido los datos (frame y nbin):
 - Lee el canal y la máscara contenidos en la tabla de la sesión para realizar los análisis.
 - De ellos toma una parte del canal en función de la máscara que dependerá de qué frame sea la indicada.
 - Realiza el análisis estadístico y el cálculo del histograma
 - Fija la variable "var" con el valor 1
- Envía a la página "stat_grafic.html" un diccionario con toda la información. Si la "var" es 1 significa que la página va a mostrar los resultados, en caso de que no lo sea mostrará el mensaje adecuado.

Sleep Analysis

Realiza el análisis del sueño

Función "*sleep_graphic*"

- Comprueba los datos que le han llegado
- Si no ha llegado ningún dato:
 - Genera una variable "var" con valor 2
- Si se han recibido los datos (el intervalo de sueño)
 - Lee el registro original, pero el canal que está indicado en la tabla de sesión.
 - Del registro toma sólo el intervalo de sueño.
 - Realiza el análisis del sueño
 - Fija la variable "var" con el valor 1
- Envía a la página "sleep_grafic.html" un diccionario con toda la información. Si la "var" es 1 significa que la página va a mostrar los resultados, en caso de que no lo sea mostrará el mensaje adecuado.

Cosinor Analysis

Realiza el análisis del Cosinor

Función "*cosinor_graphic*"

- Comprueba los datos que le han llegado
- Si no ha llegado ningún dato:
 - Genera una variable "var" con valor 2
- Si se han recibido los datos (los enviados mediante el formulario de la herramienta del cosinor)
 - Lee el canal contenido en la tabla de la sesión para realizar los análisis.
 - Realiza el análisis del cosinor
 - Fija la variable "var" con el valor 1
- Envía a la página "cosinor_grafic.html" un diccionario con toda la información. Si la "var" es 1 significa que la página va a mostrar los resultados, en caso de que no lo sea mostrará el mensaje adecuado.

Symbolic Analysis

Realiza el análisis estadístico

Función "*symbol_graphic*"

- Comprueba los datos que le han llegado
- Si no ha llegado ningún dato:
 - Genera una variable "var" con valor 2
- Si se han recibido los datos del formulario de "Symbolic Analysis"
 - Lee el canal y la máscara contenidos en la tabla de la sesión para realizar los análisis.
 - De ellos toma una parte del canal en función de la máscara que dependerá de qué frame sea la indicada.
 - Realiza el análisis simbólico
 - Fija la variable "var" con el valor 1

- Envía a la página "symbol_grafic.html" un diccionario con toda la información. Si la "var" es 1 significa que la página va a mostrar los resultados, en caso de que no lo sea mostrará el mensaje adecuado.

CTM

Realiza la medida de la tendencia central

Función "*ctm_graphic*"

- Comprueba los datos que le han llegado
- Si no ha llegado ningún dato:
 - Genera una variable "var" con valor 2
- Si se han recibido los datos (frame y rho)
 - Lee el canal y la máscara contenidos en la tabla de la sesión para realizar los análisis.
 - De ellos toma una parte del canal en función de la máscara que dependerá de qué frame sea la indicada.
 - Realiza el CTM
 - Fija la variable "var" con el valor 1
- Envía a la página "ctm_grafic.html" un diccionario con toda la información. Si la "var" es 1 significa que la página va a mostrar los resultados, en caso de que no lo sea mostrará el mensaje adecuado.

Classifier

Realiza la clasificación automática del registro

Función "*classifier_graphic*"

- Comprueba los datos que le han llegado
- Si no ha llegado ningún dato:
 - Genera una variable "var" con valor 2
- Si se han recibido el tipo de clasificador y los parámetros que éste necesita (actualmente sólo hay implementado un clasificador, pero está preparado para poder añadir más en un futuro):
 - Si el clasificador es KNN:
 - Comprueba que se pueden calcular las características explicadas en el apartado 2.9
 - En caso afirmativo realiza la clasificación y fija "var" con 1
 - Si esto no es posible, "var" será igual a 2
 - En caso de que no sea.
 - Fija "var" igual a 2
- Envía a la página "knn_grafic.html" un diccionario con toda la información. Si la "var" es 1 significa que la página va a mostrar los resultados, en caso de que no lo sea mostrará el mensaje adecuado.

4.6 DOWNLOAD RESULTS

Ofrece al usuario la posibilidad de realizar todos los cálculos vistos en "Application" pero pulsando un solo botón. Los resultados en este caso no son gráficas o tablas, sino una hoja de cálculo que contiene cada una de las salidas de los algoritmos, así como los parámetros que han sido elegidos para su cálculo.

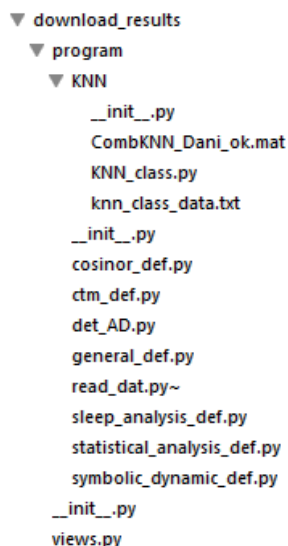


Figura 26: Árbol de "download_results"

La Figura 26 muestra el sub-árbol de archivos de esta aplicación (muy similar al visto en la Figura 18). Puede verse que todos los programas destinados al cálculo de algoritmos de sección 2 están repetidos. La razón de utilizar dos programas que hagan lo mismo reside en que para "Application" se necesitaban datos para la representación de las gráficas y ahora, para "Download Results" no son necesarios, haciendo más rápido el cálculo de los mismos y ocupando menos memoria en el proceso.

Esta herramienta no cuenta con un modelo, ya que toma los datos del registro directamente de la base de datos utilizando el modelo establecido en "Application". Además, esta aplicación es "sólo de lectura", lo que significa que no modificará ningún valor de la base de datos.

4.6.1 DISEÑO DE LA INTERFAZ Y PETICIONES AL SERVIDOR.

El diseño de esta segunda aplicación está encaminado a que el usuario pueda conseguir resultados realizando tres sencillos pasos (Figura 27):

- 1) Elegir el registro que se tomará de referencia para el cálculo.
- 2) Rellenar el formulario que permite parametrizar los algoritmos.
- 3) Descargar la hoja de cálculo generada.

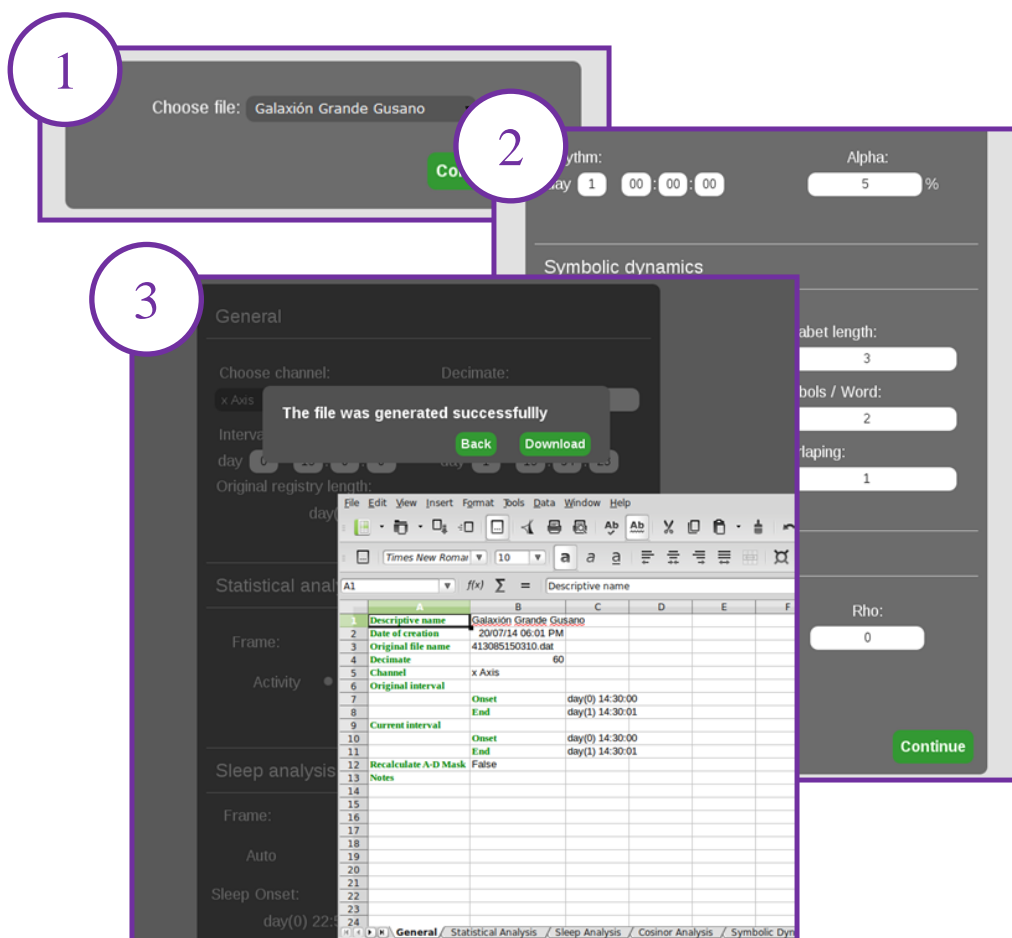


Figura 27: Diseño "Download Results"

Su sistema de plantillas está compuesto por una única página como muestra la Figura 28

```

▼ download_results
  download_results.html
  
```

Figura 28: Árbol de plantillas de "Download Results"

Cuenta con los mismos formularios que "Application", pudiéndose realizar los mismos cálculos en ambas. También cuenta con unas características similares:

- *Detección de errores*: si alguna parte del formulario está cumplimentada, el error será ensalzado automáticamente y el botón que permite continuar se deshabilitará.
- *Valores por defecto*: todas las áreas están completadas con los valores más comunes en este tipo de algoritmos, de forma que el usuario sólo tendrá que cambiar algunos parámetros.

El diseño de la hoja de cálculo (Figura 29) es simple, se generan las siguientes pestañas:

- **General**: Contiene las características generales del registro como pueden ser el diezmo, el nombre descriptivo o el canal utilizado.

- Statistical Analysis: contiene los datos y los resultados del análisis estadístico.
- Sleep Analysis: contiene los datos y los resultados del análisis del sueño.
- Cosinor Analysis: contiene los datos y resultados del cosinor.
- Symbolic Dynamic: contiene los datos y resultados de la dinámica simbólica.
- CTM: contiene los datos y los resultados del CTM.
- Classifier: contiene los datos y los resultados del clasificador.

Para hacer la lectura de los datos más sencilla, los títulos se han puesto en color verde, mientras que los resultados son de color negro.

	A	B	C	D	E	F	G
1	Descriptive name	Registro 1					
2	Date of creation	08/07/14 12:58 PM					
3	Original file name	384632090310.dat					
4	Decimate	60					
5	Channel	x Axis					
6	Original interval						
7		Onset	day(0) 13:00:00				
8		End	day(0) 01:00:00				
9	Current interval						
10		Onset	day(0) 13:00:00				
11		End	day(1) 13:54:23				
12	Notes						
13	Notas del Registro 1						
14							

Figura 29: Ejemplo de una hoja de cálculo

4.6.2 VISTA

Para acceder a esta aplicación es necesario estar registrado. Se compone de tres funciones:

Función "*download_results*". Es la función encargada de mostrar los datos que deben visualizarse.

- Envía los nombres descriptivos de los registros que pertenecen al usuario y se encuentran en la base de datos (var = 1).
- Cuando alguno de ellos sea elegido, retornará los datos necesarios para completar los formularios de la web (var = 2).
- Envía a la página "*download_results.html*" un diccionario con toda la información. Si la "var" es 1 significa que la página cargará la selección de registros, en caso contrario, mostrará el formulario completo

Función "*generate_results*". Se encarga de calcular todos los resultados y guardarlos en una hoja de cálculo. Los parámetros de los algoritmos y la forma en que se calculan son análogas a las utilizadas en "Application", por lo que no van a ser repetidas.

- Recibe la información con el registro elegido y todos los parámetros para calcular los algoritmos.
- Calcula cada uno de los algoritmos.

- Crea una hoja de cálculo.
- Genera una pestaña dentro de la hoja de cálculo para cada una de las partes (General, Statistical Analysis, etc.) y escribe en ella tanto los resultados como las opciones utilizadas para obtenerlos.
- Guarda el archivo en el directorio "tmp" con un nombre compuesto por el nombre del usuario, el año, el mes, el día, la hora, los minutos y los segundos actuales, separando estos por una barra baja (Ejemplo: "dani_2014_8_31_21_24_9.xls").
- Envía la ruta completa si la hoja ha sido generada con éxito.

Función *"get_results"*. Permite la descarga de la hoja.

- Recibe la ruta absoluta de la hoja de cálculo.
- Comprueba que el nombre del archivo comienza por el nombre del usuario.
- Si es correcto:
 - Copia la hoja completa.
 - Elimina la que existe en el directorio "tmp".
 - Sirve el archivo.
- En caso de no cumplirse, responde con un mensaje de error.

4.7 CONTACT

Es una aplicación simple que consta de un formulario destinado a cualquier visitante de la web, y que se envía al administrador mediante SMTP (Simple Mail Transfer Protocol).

Su directorio base está compuesto por los archivos mostrados en la Figura 30 . En esta se muestra un archivo aún no comentado.

```

▼ contact
  __init__.py
  forms.py
  views.py

```

Figura 30: Árbol de "Contact"

"forms.py" es el nombre establecido por Django para alojar la creación de formularios en el lado del servidor. De la misma forma que puede generar tablas, Django puede generar formularios y su corrección con muy pocas líneas de código.

```

#CONTACT
class contact_form(forms.Form):
    name = forms.CharField(max_length=100,required=True)
    email = forms.EmailField(required=True)
    subject = forms.CharField(max_length=100,required=True)
    message = forms.CharField(max_length=2000,required=True,widget=forms.Textarea)

```

Figura 31: Código fuente de "contact/forms.py"

El código fuente visto en la Figura 31 es capaz de generar un formulario con cuatro apartados y con las restricciones necesarias para saber si está completado correctamente o no.

La herramienta que nos permite hacer esto es el paquete "forms" de Django. El formulario de contacto está formado por:

- Nombre: debe insertarse el nombre (máximo de 100 caracteres).
- Correo: debe introducirse un correo válido.
- Asunto: debe rellenarse con el carácter del mensaje (máximo de 100 caracteres).
- Mensaje: debe escribirse el mensaje que se desea hacer llegar (máximo de 2000 caracteres).

Todos los campos cuentan con la declaración "required=True", que hace que el campo sea obligatorio. Por otro lado "max_length" define el número de caracteres máximo que debe tener ese apartado.

Para esta aplicación tampoco es necesario definir un modelo, ya que no necesita ningún dato adicional para su ejecución.

4.7.1 DISEÑO DE LA INTERFAZ Y PETICIONES AL SERVIDOR

El diseño está compuesto por el formulario de la Figura 32.

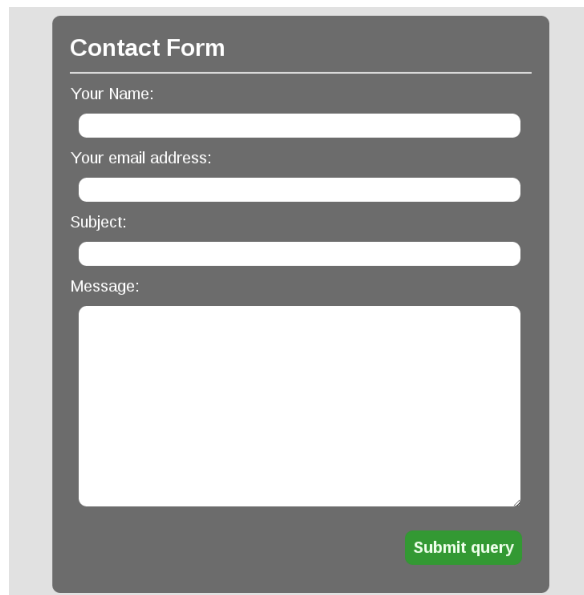


Figura 32: Formulario de contacto

El código se encuentra en el archivo "contact.html" (Figura 33).

```
▼ contact
  contact.html
```

Figura 33: Árbol de plantillas de "contact"

4.7.2 VISTA

La única función necesaria para esta aplicación es la llamada "contact", que sigue el siguiente algoritmo:

Toma el nombre del usuario que esté registrado en ese momento, si no hay ninguno, se toma como anónimo.

- Comprueba si en la información enviada existe un formulario
- Si existe:
 - Comprueba que el formulario sea correcto.
 - Si este es correcto:
 - Toma la información del formulario y la envía.
 - Indica a la página que muestre el mensaje de que la información ha sido enviada correctamente.
 - Si no lo es:
 - Retorna a la página los errores cometidos
- Si no existe:
 - Muestra el formulario sin rellenar ("contact.html")

4.8 LOG-IN/LOG-OUT

Es la aplicación encargada de gestionar el inicio y cierre de sesiones de cada usuario. Para su implementación se ha hecho uso de la aplicación de autenticación de Django (auth), mediante la cual es posible identificar a usuarios que se encuentren ya registrados en la base de datos.

La primera vez que es utilizada por un usuario se crea la información básica requerida en la tabla de sesión. Su raíz puede verse en la Figura 34.

```
▼ log_in_out
  __init__.py
  admin.py
  models.py
  views.py
```

Figura 34: Árbol de "log_in_out"

4.8.1 MODELO

Con esta aplicación se introduce una nueva tabla en la base de datos, que será la encargada de almacenar el nombre del usuario, la fecha en la que se ha realizado el registro y la IP desde donde se ha realizado. Su uso es meramente estadístico y sirve para conocer la afluencia de la web y los lugares desde donde se accede (Figura 35).

```
class access_list(models.Model):
    user_name = models.ForeignKey(User)
    user_ip = models.IPAddressField()
    date = models.DateTimeField('log in date', auto_now=True)
    def __unicode__(self):
        return self.user_ip
```

Figura 35: Modelo para el registro de las sesiones

4.8.2 DISEÑO DE LA INTERFAZ

Su diseño es simple. Como se muestra en la Figura 36, el formulario consta de dos campos, el de nombre y el de contraseña. Los caracteres insertados en el apartado de contraseña aparecerán como puntos.

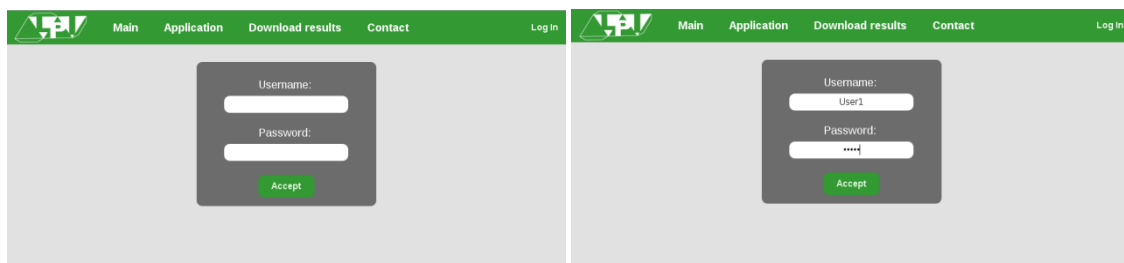


Figura 36: Formulario de inicio de sesión

El acceso a esta aplicación se ha insertado en la parte superior derecha de la cabecera, de forma que siempre esté presente. Cuando el registro se ha efectuado, el contenido cambia, mostrando ahora el nombre propietario de la sesión actual y dando la opción de cerrar sesión (Figura 37).

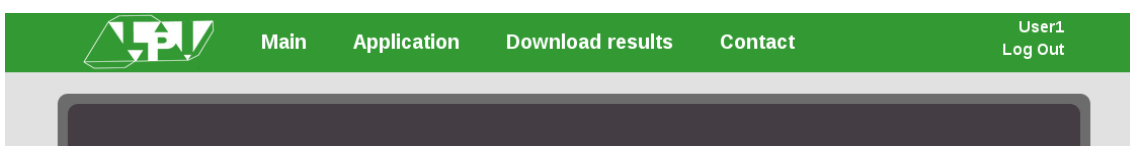


Figura 37: Usuario registrado

El directorio que conforma el sistema de plantillas de esta aplicación puede verse en la Figura 38.

```
▼ log_in_out
  log_in.html
```

Figura 38: Esquema de las plantillas de "Log-in/Log-out"

4.8.3 VISTA

En la vista se muestra el algoritmo para realizar el inicio y el cierre de sesión. Consta de dos funciones:

Función *"log_in"*. Se encarga del inicio de sesión.

- Comprueba que el formulario de inicio de sesión esté bien relleno.
- Si lo está:
 - Comprueba que el usuario y la contraseña exista y estén activos.
 - Si no está activo, devuelve el mensaje correspondiente.
 - Si no es correcta, retorna el error correspondiente.
 - En caso de que sea correcta y esté activa.
 - Comprueba que sea la primera vez, que ha iniciado sesión.
 - Si es la primera vez, genera la tabla de sesión.
 - En caso de que no sea la primera vez, actualiza el campo de "session_key", además comprueba si existe alguna sesión creada con esa misma "session_key", y si es así, la cierra.
 - Una vez iniciada la sesión correctamente, envía al usuario por defecto a la aplicación principal.
- Si no está completado correctamente, volverá a pedirle la información.

Función *"log_out"*. Se encarga del cierre de sesión.

- Cierra la sesión del usuario registrado.

4.9 OTHER

Esta aplicación se compone de dos herramientas que sirven para complementar las aplicaciones principales:

- Página de presentación: es una página en la que se realiza una breve descripción de la aplicación y se da acceso al manual de usuario de la aplicación.
- Página de carga: es la página que se muestra cuando se está esperando a que una página reciba la respuesta del servidor o finalice su cargarse.

No existen características especiales en estas webs en el lado del servidor, ya que son páginas totalmente estáticas y no necesitan acceder a la base de datos para su ejecución.

Su directorio raíz puede verse en la Figura 39

```
▼ other
  __init__.py
  views.py
```

Figura 39: Árbol de "other"

4.9.1 DISEÑO DE LA INTERFAZ

El árbol de plantillas de esta aplicación puede verse en la Figura 40. En ella se muestran 5 archivos que veremos a continuación.

```
▼ other
  loading_1.html
  loading_2.html
  main.html
  processing_data.html
  uploading_file.html
```

Figura 40: Árbol de plantillas de "other"

La presentación de la web es un aspecto importante de cara al público que visite la página. Para ello se ha creado una página principal (main.html) que describe la aplicación (Figura 41).

Desde esta página también está disponible la descarga del manual de usuario y un acceso directo a la página de contacto.

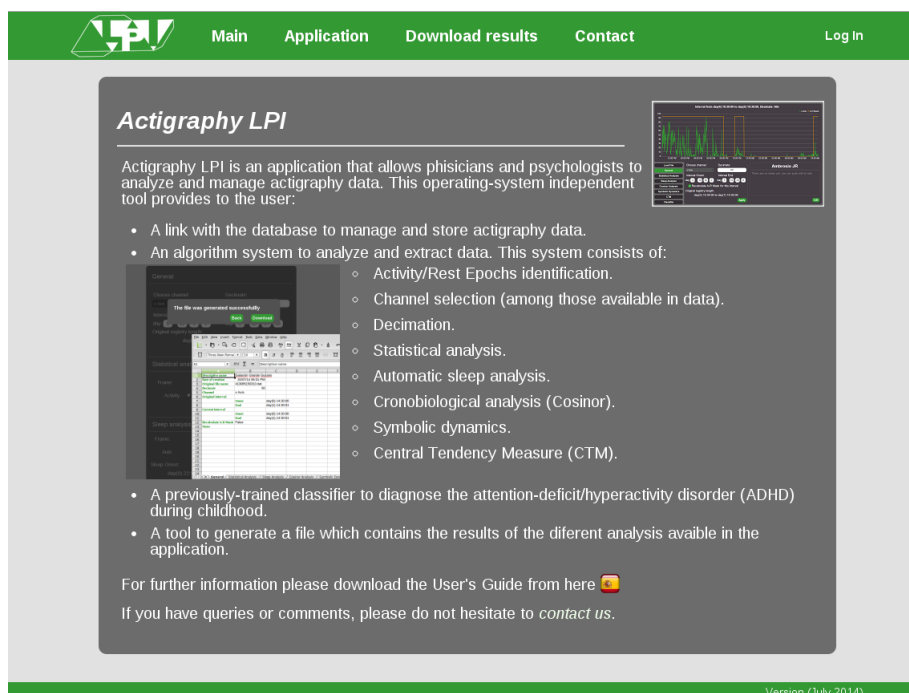


Figura 41: Página principal de Actigraphy LPI

En este árbol, hay cuatro páginas más que están dedicadas meramente al uso estético. Se componen por un elemento de HTML5 llamado "canvas" con el cual puede crearse contenido gráfico. La Figura 42 muestra algunos ejemplos.



Figura 42: Ejemplo de animación de espera

4.9.2 VISTA

Las vistas utilizadas para el control de estas páginas es muy simple, cada vez que el servidor ejecute las funciones, éstas retornarán la página web correspondiente y su contexto estático.

Función "*main*"

- Retorna la página "main.html"

Función "*loading_1*"

- Retorna la página "loading_1.html"

Función "*loading_2*"

- Retorna la página "loading_2.html"

Función "*uploading_file*"

- Retorna la página "uploading_file.html"

Función "*processing_data*"

- Retorna la página "processing_data.html"

4.10 ADMIN

Esta aplicación pertenece a Django y crea una interfaz de administrador que permite gestionar cualquier dato que se encuentre en la base de datos e incluir nuevas entradas.

La forma de acceder a ella es desde la URL principal, añadir "/admin" en ella y pulsar "intro". Inmediatamente aparecerá un entorno similar al de la Figura 43

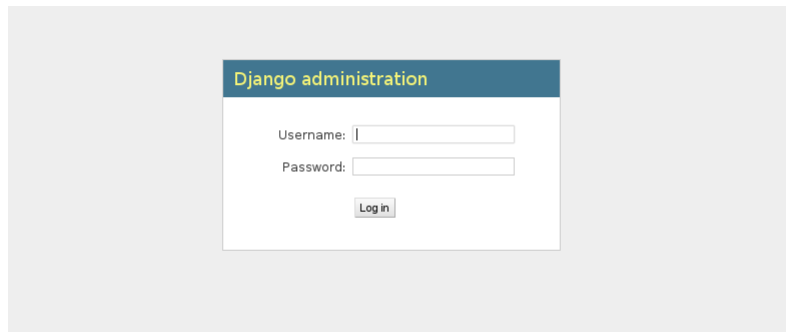


Figura 43: Formulario de inicio de "Admin"

El usuario y la contraseña por defecto en esta aplicación son (Deben ser cambiadas en la fase de producción):

- Username: dani
- Password: debian64

Una vez introducidos, la interfaz de administrador muestra multitud de opciones (Figura 44).

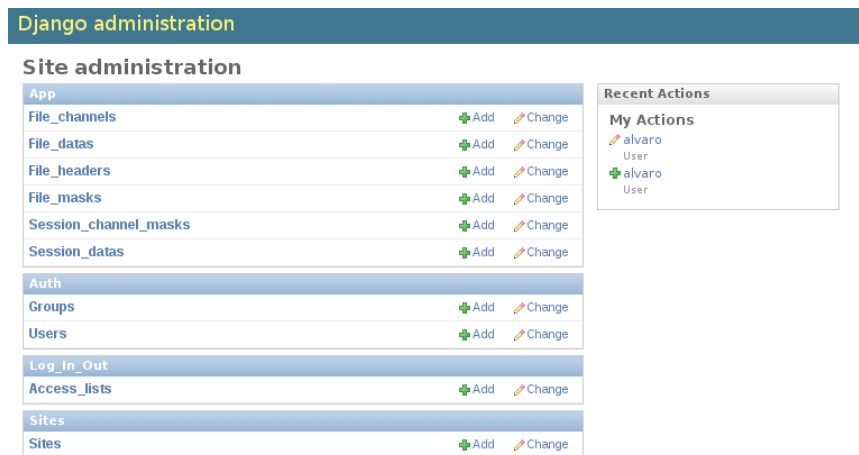


Figura 44: Interfaz de administrador

Desde ellas puede accederse a cada una de las tablas creadas mediante los modelos, además de a otras nuevas tablas creadas por las aplicaciones instaladas de Django (como "Auth" que permite crear nuevos usuarios).

El algoritmo utilizado por "Auth" para la codificación de las contraseñas es el PBKDF2 (Password-Based Key Derivation Function 2), y el número de iteraciones que usa es de 1000.

5 CONCLUSIONES Y LÍNEAS FUTURAS

En el siguiente capítulo se expresan las ideas y conclusiones propias tras la realización de este proyecto, comentando desde el punto de vista personal cuáles son los puntos fuertes del mismo. En el apartado de líneas futuras se dará una opinión de cómo este proyecto podría evolucionar y/o completarse.

5.1 CONCLUSIONES

El TDAH es un trastorno muy común en la infancia que, si no es diagnosticado y tratado, suele traer consigo otros problemas de enorme relevancia en la vida cotidiana como ansiedad, tics, trastorno del estado de ánimo, etc., y puede incluso que se mantenga durante la adolescencia y la edad adulta.

La actigrafía es un método no invasivo en virtud del cual se registran los movimientos del sujeto en su medio de vida habitual por lo que resulta una herramienta muy cómoda para analizar su comportamiento. Asimismo, es conocido que la actigrafía posee un notable potencial diagnóstico para el trastorno que acabamos de mencionar.

Por otra parte, las tecnologías web ofrecen la posibilidad de crear aplicaciones de todo tipo de manera sencilla y ubicua. En consecuencia, hoy en día es posible la reproducción de música y de vídeo, realizar trabajos ofimáticos y muchas otras tareas, tanto comunes como profesionales, requiriéndose únicamente un navegador web para tal fin.

Mediante este proyecto se ha hecho posible el análisis on-line de un registro actigráfico, en el que se incluye la visualización de la señal, la determinación de los intervalos de vigilia-descanso, la obtención de descriptores numéricos de la señal y, finalmente, la toma de una decisión, según un sistema de clasificación previamente entrenado, de si el caso bajo estudio corresponde a un paciente sano o afectado por TDAH. Además, proporcionamos como salida los datos del estudio para permitir un análisis más pormenorizado desde otras plataformas.

Esperamos que esta herramienta sea del agrado de los profesionales del sector, que la comodidad en su manejo (objetivo perseguido a lo largo del desarrollo) se traduzca en un uso efectivo de la misma y que, incluso, se pueda establecer un círculo de profesionales alrededor de ésta con un fin común, a saber, el diagnóstico correcto del TDAH en la infancia mediante la actigrafía, un método objetivo, económico y rápido.

En mi opinión, con esta aplicación se consigue poner un medio para un fin y supone una aportación en el desarrollo de este campo.

5.2 LÍNEAS FUTURAS

Gracias a las múltiples posibilidades que ofrece la web, la aplicación creada puede tener un gran número de posibles líneas de trabajo. A continuación se exponen las más relevantes.

Por supuesto, varias resultan evidentes, como:

- Aumentar el soporte para una mayor cantidad de actígrafos y modos.
- La implementación de más herramientas para la obtención de características.
- La implementación de nuevos clasificadores.

Pero, como se ha visto a lo largo de esta memoria, ahora estamos asistiendo a la segunda generación web, la Web 2.0, la web de las redes sociales. Esta aplicación podría contener también un foro, en el cual distintos profesionales pudieran comentar sus experiencias o pedir opinión a otros y proponer mejoras de la web, dejando la puerta abierta a que terceros aporten nuevo código.

Otro camino alternativo sería pedir la colaboración de los usuarios para que proporcionasen registros actigráficos y su diagnóstico con el fin de crear una gran base de datos y poder así ofrecer mejores herramientas y clasificadores.

¿Por qué parar aquí?. El diagnóstico del TDAH es sólo una posible aplicación de la actigrafía. La utilidad creada podría extenderse también al diagnóstico de otras patologías, como cólicos infantiles o el insomnio.

6 BIBLIOGRAFÍA

1. *Best practices in adult ADHD: Epidemiology, impairments and differential diagnosis.* **L.A. Adler, T.J. Spencer, M.A. Stein, and J.H. Newcorn.** 2008, *CNS Spectr.*, Vol. 13, págs. 4-5.
2. *Current strategies in the diagnosis and treatment of childhood attention-deficit/hyperactivity disorder.* **R. Rade, L. MacCaulew, and E.C. Callen.** 2009, *Am. Fam. Physician*, Vol. 79, págs. 657-665.
3. *Practice parameter for the assessment and treatment of children and adolescents with attention-deficit/hyperactivity disorder.* **Pliszka, S.** 2007, *J. Am. Acad. Child. Adolesc. Psychiatry*, Vol. 46, págs. 894-921.
4. *Attention deficit hyperactivity disorder in 2010.* **D. Efrom, P. Hazell, and V. Anderson.** 2010, *J. Paediatr. Child Health*, Vol. Dec. 2010.
5. *Transtorno por déficit de atención e hiperactividad: Un problema actual.* **M.D. García-García, L.M. Prieto-Tato, J. Santos-Borbujo, L. Monzón-Corral, A. Hernández-Fabián, and L. San Feliciano-Martín.** Barcelona : s.n., 2008, *An. Pediatr.*, Vol. 69, págs. 244-250.
6. **workgroup, DSM-IV-TR.** *The Diagnostic and Statistical Manual of Mental Disorders, Fourth Edition.* Fourth Edition. s.l. : American Psychiatric Association, 2000.
7. *Attention deficit disorder in reading disabled twins: evidence for a genetic etiology.* **J.J. Guillis, J.W. Gilger, B.F. Pennington, and j.C. DeFries.** 1992, *Abnorm. Child. Psychol.*, Vol. 20, págs. 303-315.
8. *Neuroscience of attention deficit/hyperactivity disorder: The search for endophenotypes.* **Tannochk, F.C. Castellanos and R.** 2002, *Nat. Neurosci*, Vol. 3, págs. 617-628.
9. *ADHD genetics: 2007 update.* **Devoto, J. Elia and M.** 2007, *Curr. Psychiatry Rep.*, Vol. 9, págs. 434-439.

10. *Neurobiology of attention-deficit hyperactivity disorder*. **Biederman, S.V. Faraone and J.** 1998, *Bio. Psychiatry*, Vol. 44, págs. 951-958.
11. *Association of Dopamine D4 Receptor (DRD4) Gene with Attention-Deficit/Hyperactivity Disorder (ADHD) in a High-Risk Community Sample: a Longitudinal Study from Birth to 11 Years of Age*. **M. El-Faddagh, M. Laucht, A. Maras, L. Vöhringer, and M. H. Schmidt.** 2004, *J. Neural. Transm.*, Vol. 111, págs. 883-889.
12. *Frontostriatal Connectivity and Its Role in Cognitive Control in Parent-Child Dyads with ADHD*. **B.J. Casey, J.N. Epstein, J. Buhle, C. Liston, M.C. Davidson, S.T. Tonev, J. Spicer, S. Niogi, A.J. Millner, A. Reiss, A. Garrett, S.P. Hinshaw, L.L Greenhill, K.M. Shafritz, A. Vitolo, L.A. Kotler, M.A. Jarrett, and G. Glover.** 2007, *Am. J. Psychiatry*, Vol. 164, págs. 1729-1736.
13. *Test Anxiety: The State of the Art (Perspectives on Individual Differences)*. **Zeidner, M.** 1998, Kluwer Academic Publishers.
14. **M. Green, M. Wong, D. Atkins, J. Taylor, and M. Feinleib.** *Diagnosis of Attention-Deficit/Hyperactivity Disorder. Technical Review, Number3.* s.l. : Agency for Health Care Policy and Research, U.S. Department of Health and Human Services, 1999.
15. *Telayed Circadian Rhythm in Adults with Attention-Deficit/Hyperactivity Disorder and Chronic Sleep-Onset Insomnia*. **M.M Van-Veen, J.J.S. Kooij, M.M. Boonstra, M.C.M.Gordijn, and E.J.W. Van-Someren.** 2010, *Biol. Psychiatry*, Vol. 67, págs. 1091-1096.
16. *Parameterization-Invariant Shape Comparisons of Anatomical Surfaces*. **S. Kurtek, E. Klassen, Zhaohua Ding, S.W. Jacobson, J.L. Jacobson, M.J. Avison, and A. Srivastava.** 2011, *IEEE Trans. Med. Imag.*, Vol. 30, págs. 849-858.
17. *Attention-Deficit /Hyperactivity Disorder: A Preliminary Diffusion Tensor Imaging Study*. **M. Ashtari, S. Kumra, S.L. Bhaskar, T. Clarke, E. Thaden, K.L. Cervellione, J. Rhinewine, J.M. Kane, A. Adesman, R. Milanaik, J. Maytal, A. Diamond, P. Szeszko, and B.A. Ardekani.** 2005, *Biol. Psychiatry*, Vol. 57, págs. 448-455.
18. *The Restless Brain: Attention-Deficit Hyperactivity Disorder, Resting-State Functional Connectivity, and Intrasubject Variability*. **F.X. Castellanos, C. Kelly, and M.P. Milham.** 2009, *Can. J. Psychiatry*, Vol. 54, págs. 665-672.
19. *White-Matter Abnormalities in Attention Deficit Hyperactivity Disorder: A Diffusion Tensor Imagin Study*. **T.J. Silk, N. Vance, A. Rinehart, J.L Bradshaw, and R. Cunningtin.** 2009, *Hum. Brain Map.*, Vol. 30, págs. 2757-2765.
20. *Positron Emission Tomography Neuroimaging for a Better Understanding of the Biology of ADHD*. **Zimmer, L.** 2009, *Neuropharmacology*, Vol. 57, págs. 601-607.
21. *A Neuro-Behavioral Test and Algorithms for Screening and Evaluating Therapy in ADHD*. **M. Modarres-Zadeh, Xueyan Xu, A. Jaromczyk, and S. Senders.** 2005, *Proceedings of the*

International Annual Conference of the IEEE-Engineering in Medicine and Biology Society, 2005., págs. 2486-2489.

22. *Auxiliary Neurofeedback System for Diagnostic of Attention Deficit Hyperactivity Disorder.* **J.A. Gutierrez Gnecci, J.C. Herrera Garcia, and J. de Dios Ortiz Alvarado.** 2007, Proceedings of the Electronics, Robotics and Automotive Mechanics Conference, 2007., págs. 135-138.

23. *Analysis of Attention Deficit Hyperactivity Disorder in EEG Using Wavelet Transform and Self Organizing Maps.* **S. Hyun-Lee, B. Abibullaev, Won-Seok Kang, Yunhee Shin, and Jinung An.** 2010, Proceedings of the Control Automation and Systems., págs. 2439-2442.

24. *Multiscale entropy analysis of attention related EEG based on motor imaginary potentia.* **D. Ming, M. Zhang, Y. Xi, H. Qi, Y. Hu, and K.D.K. Luk.** 2009, Proceedings of the International Conference on Computational Intelligence for Measurement Systems and Applications, págs. 24-27.

25. *Analysis of Propagation of Multi-Channel EEG in the Test of Sustained Attention.* **Y. Nan, J. Wang, S.A. Xue, H. Sheng, Y. Jiao, and J. Wang.** 2010, Proceedings of the International Annual Conference of the IEEE-Engineering in Medicine and Biology Society, págs. 1666-1669.

26. *Atypical EEG Beta Asymmetry in Adults with ADHD.* **T.S. Hale, S.L. Smalley, P.D. Walshaw, G. Hanada, J. Macion, J.T. McCracken, J.J. McGough, and S.K. Loo.** 2010, Neuropsychologia, Vol. 48, págs. 3532-3539.

27. *Classification of ADHD/Normal Participants Using Frequency Features of ERP's Independent Comonents.* **F. Ghassemi, M.H. Moradi, M. Tehrani-Doost, and V. Abootalebi.** 2010, Proceedings of the Complex Medical Engineering, págs. 1-4.

28. *Evaluation Of Feature Extraction Techniques On Event-Related Potentials For Detection Of Attentiion-Deficit/Hyperactivity Disorder.* 2010, Proceedings of the International Annual Conference of the IEEE-Engineering in Medicine and Biology Society, págs. 851-854.

29. *Low Frequency Phase Synchronisation Analysis of MEG Recordings from Children with ADHD and Controls Using Single Channel ICA.* **C. Demanuele, C.J. James, E.J.S. Sonuga-Barke, and A. Capilla.** 2008, Proceedings of the International Conference on Advances in Medical, Signal and Information Processing, págs. 1-4.

30. *Practice Parameters for the Indications for Polysomnography and Related Procedures.* **A.L. Chesson, R.A. Ferber, J.M. Fry, M. Grigg-Damberger, K.M. Hartse, T.D. Hurwitz, S. Johnson, M. Littner, G.A. Kader, G. Rosen, R.B. Sangai, W. Schmidt-Nowara, and A. Sher.** 1997, Sleep, Vol. 20, págs. 406-422.

31. *Instability of Sleep Patterns in Children With Attention-Deficit/Hyperactivity Disorder.* **R. Gruber, A. Sadeh, and A. Raviv.** 2000, J. Am. Acad. Child. Adolesc. Psychiatry, Vol. 39, págs. 495-501.

32. *Sleep Disorders in Children with Attention-Deficity/Hyperactivity Disorder Recorded Overnight by Video-Polysomnography*. **R. Silvestri, A. Gagliano, I. Aricò, T. Calarese, C. Cedro, O. Bruni, R. Concurso, E. Germanó, G. Gervasi, R. Siracusano, G. Vita, and P. Bramanti.** 2009, *Sleep Med*, Vol. 10, págs. 1132-1138.
33. *Sleep in children with attention-dedicit/hyperactivity disorder: A meta-analysis of polysomnographic studies*. **A. Sadeh, L. Pergamin, and Y. Bar-Haim.** 2006, *Sleep Med. Rev.*, Vol. 10, págs. 381-398.
34. *Sleep in Children With Attention-Deficit/Hyperactivity Disorder: Meta-Analysis of Subjective and Objective Studies*. **S. Cortese, S. Faraone, E. Konofal, and M. Lecendreux.** 2009, *J. Am. Acad. Child. Adolesc. Psychiatry*, Vol. 48, págs. 894-908.
35. *Objective Measurement of Hyperactivity and Attentional Problems in ADHD*. **M.H. Teicher, Y. Ito, C.A. Glod, and N.I. Barber.** 1996, *J. Am. Acad. Child. Adolesc. Psychiatry*, Vol. 35, págs. 334-342.
36. *Quantitative Assessment of Fidgeting in Attention-Deficit/Hyperactivity Disorder*. **M.H. Teicher, Y. Ito, C.A. Glod, and P. Wallace.** 1993, *Am. Acad. Child Adolescence Psychiatry*, Vol. 9, págs. 42-43.
37. *Actigraphy and Motion Analysis: New Tools for Psychiatry*. **Teicher, M.H.** 1995, *Harvard Rev. Psychiatry*, Vol. 3, págs. 18-35.
38. *An Objective Measure of Hyperactivity*. **Reisman, J.L. Shulman and J.M.** 1959, *Am. J. Ment. Defic.*, Vol. 64, págs. 455-456.
39. *Physical Activity Assessment: A Review of Methods*. **Freedson, E.L. Melanson Jr. and P.S.** 1996, *Crit. Rev. Food Sci. Nutr.*, Vol. 36, págs. 385-396.
40. *Does Actigraphy Differentiate ADHD Subtypes in a Clinical Research Setting?* **A.V. Dane, R.J. Schachar, and R. Tannok.** 2000, *J. Am. Acad. Child. Adolesc. Psychiatry*, Vol. 39, págs. 752-760.
41. *Actigraph Assessment of Rest-Activity Disturbances in Psychiatric Disorders*. **M.H. Teicher, C.E. McGreenery, and K. Ohashi.** 2006. *Proceedings of the World Congress on Psychosomatic Medicine, 2006*. Vol. 25 of 1287 of *International Congress Series*, págs. 32-37.
42. *Locomotor Activity and the Diagnosis of ADHD*. **M.H. Teicher, C.A. Glod, K. Pahlavan, D. Harper, E. Magnus, and F. Wren.** 1991, *Am. Psychiatr Assoc. Abstr.*, Vol. 141, pág. 98.
43. *High-Resolution Actigraphic Analysis of ADHD: A Wide Range of Movement Variability Observation in Three School Courses - A Pilot Study*. **H.J. Kam, K. Lee, S.M. Cho, Y.M. Shin, and R.W. Park.** 2011, *Healthc. Inform. Res.*, Vol. 17, págs. 19-34.
44. *Ambulatory Activity Monitoring: Progress in Measurement of Activity, Posture, and Specific Motion Patterns in Daily Life*. **J.J. Bussmann, U.W. Ebner-Priemer, and J. Fahrenberg.** 2009, *Eur. Psychol.*, Vol. 14, págs. 142-152.

45. *Activity-based Sleep-Awake Identification: an Empirical Test of Methodological Issues*. **A. Sadeh, K. Sharkey, and M. Carskadon**. 1994, *Sleep*, Vol. 17, págs. 201-207.
46. *Activity-Based Assessment of Sleep-Wake Patterns During the 1 st Year of Life*. **A. Sadeh, C. Acebo, R. Seifer, S. Aytur, and M.A. Carskadon**. 1995, *Infant Behav. Dev.*, Vol. 19, págs. 329-337.
47. *Sleep Estimation from Wrist Movement Quantified by Different Actigraphic Modalities*. **G. Jean-Louis, D. Kripke, W. Mason, J. Elliot, and S. Youngstedt**. 2001, *J. Neurosci. Methods*, Vol. 105, págs. 185-191.
48. *Automatic Sleep/Wake Identification from Wrist Activity*. **R.J. Cole, D. Kripke, W. Gruen, D.J. Mullaney, and J.C. Gillin**. 1992, *Sleep*, Vol. 15, págs. 461-469.
49. *The Role and Validity of Actigraphy in Sleep Medicine: an Update*. **Sadeh, A.** 2011, *Sleep Med. Rev.*
50. *Automatic Diagnosis of ADHD Based on Nonlinear Analysis of Actimetry Registries*. **D. Martín, P. Casaseca, S. Alberola, J.A. López, F.C. Ruiz, J.M. Andrés, J.R. Garmendia and J. Ardura**. 2011, In 36th International Conference on Acoustics, Speech and Signal Processing (ICASSP'2011).
51. *Nonlinear analysis of actigraphic signals for the assessment of attention-deficit/hyperactivity disorder (ADHD)*. **D. Martín-Martínez, P. Casaseca-de-la-Higuera, S. Alberola-López, J. Andrés-de-Llano, J.A. López-Villalobos, J. Ardura-Fernández, C. Alberola-López**. 2012, *Medical Engineering & Physics*, Vol. 34, págs. 1317-1329.
52. *Methods for COSINOR-Rhythmmometry*. **W. Nelson, Y.L. Tong, and F. Halberg**. 1979, *Chronobiologia*, Vol. 6, págs. 305-323.
53. *Understanding Robust and Exploratory Data Analysis*. **Hoaglin, D., Mosteller, F. y and Tukey, J. (Ed.)**. 1983, New York: Wiley, págs. 39, 54, 62, 223.
54. **Zwillinger, D. and Kokoska, S.** *CRC Standard Probability and Statistics Tables and Formulae*. 2000.
55. *Circadian system phase - An aspect of temporal morphology*. **Halberg, F., Tong, Y. L., and Johnson, E. A.** 1967, *The Cellular Aspects of Biorhythms: Proceedings*, págs. 20-48.
56. *The application of methods of non-linear dynamics for the improved and predictive recognition of patients threatened by sudden cardiac death*. **A. Voss, J. Kurths, H. J. Kleiner, A. Witt, N. Wessel, P. Sapparin, K. J. Osterziel, R. Schurath, and R. Dietz**. 1996, *Cardiovasc. Res.*, Vol. 31, págs. 419-433.
57. *Joint symbolic dynamic analysis of beat-to-beat interactions of heart rate and systolic blood pressure in normal pregnancy*. **M. Baumert, T. Walther, J. Hopfe, H. Stepan, R. Faber, and A. Voss**. 2002, *Med. Biol. Eng. Comput.*, Vol. 40, págs. 241-245.

58. **D. Martín-Martínez* (Student Member), P. Casaseca-de-la-Higuera, J. Andrés-de-Llano J.R. Garmendia-Leiza, S. Alberola-López, J.A. López-Villalobos and C. Alberola-López (Senior Member).** *An Automatic Actigraphy-based Diagnostic Tool for the Attention Deficit/Hyperactivity Disorder.* Laboratorio de procesamiento de imagen. Valladolid : s.n., 2014.
59. *As We May Think.* **Bush, Vannevar.** 1945, The Atlantic Monthly, Vol. 176, págs. 101-108.
60. *A File Structure for the Complex, the Changing, and the Indeterminate.* **Nelson, Ted.** New York : s.n., 1965, 20th National Conference ACM, págs. 84-100.
61. *Knowledge Representation on the Web.* **Decker, S., y otros.** Aachen, Alemania : s.n., 2000, Proceedings of the 2000 International Workshop on Description Logics, págs. 89-97.
62. **Strickland, Jonathan.** *Is there a Web 1.0?* 2005.
63. **O'Reilly, Tim.** *What Is Web 2.0: Design Patterns and Business Models for the Next Generation of Software.* 2004.
64. **Markoff, John.** Entrepreneurs See a Web Guided by Common Sense. *New York Times.* 12 de Noviembre de 2006.
65. **Foundation, Python Software.** www.python.org. [En línea] 2014.
66. **Summerfield, Mark.** *Python 3.* 2009.
67. **Foundation, Django Software.** www.djangoproject.com. [En línea] 2014.
68. **Adrian Holovaty, Jacob Kaplan-Moss, et al.** *The Django Book.* s.l. : Jeremy Dunck, 2009.
69. **Montero, Sergio Infante.** *Django para perfeccionistas.* 2012.
70. **Gauchat, Juan Diego.** *El gran libro de HTML5, CSS3 y JavaScript.* s.l. : Marcombo, 2012.
71. **Richard Clark, Oli Studholme, Divya Manian, and Christopher Murphy.** *Beginning HTML5 and CSS3 - The Web Evolved.* s.l. : Apress, 2012.
72. **Tex Texin, John Cowan.** <http://www.i18nguy.com/unicode/language-identifiers.html>. [En línea] 01 de 01 de 2005.

7 ANEXO

Este anexo contiene la guía de usuario desarrollado para la aplicación web, el cuál describe detalladamente los pasos que debe realizar el usuario para la ejecución de la aplicación.

Julio 2014



Actigraphy LPI
Guía de usuario

actigrafia.lpi.tel.uva.es

Autor: Daniel Treceño

ACTIGRAPHY LPI

ACTIGRAPHY LPI

Actigraphy LPI es una aplicación web capaz de proporcionar a médicos y psicólogos una herramienta independiente de la plataforma y de alta disponibilidad que permita analizar y gestionar datos actigráficos.

Dicha aplicación cuenta con:

- Enlace con una base de datos para el almacenaje y gestión de los archivos actigráficos.
- Un sistema de algoritmos para el análisis y la extracción de características:
 - Determinación de la máscara de actividad-descanso.
 - Selección del canal entre los disponibles en el archivo.
 - Sistema de diezmado
 - Análisis estadístico
 - Análisis del sueño
 - Análisis cronobiológico (cosinor)
 - Dinámica simbólica
 - CTM (Medida de la tendencia central)
- Un clasificador entrenado previamente para el diagnóstico de pacientes con déficit de atención / hiperactividad con edades cercanas a los 6 años.
- Una herramienta mediante la cual puedan calcular y descargar datos de una forma cómoda para el usuario.

CONTENIDO

Visión general	5
Distribución de la web	5
Sistema de sesiones	7
Inicio y cierre de sesión.....	7
Aplicación	9
Estructura de la aplicación.....	9
Load File.....	10
Subir un nuevo archivo.....	10
General	13
Gráfica del canal y la máscara A-D	13
Notas	14
Statistica Analysis.....	15
Sleep Analysis.....	17
Cosinor Analysis	18
Symbolic Dynamics	20
CTM (Central Tendecy Measure)	21
Classifier	23
Descarga de resultados	24

VISIÓN GENERAL

En este manual se describen las principales características de Actigraphy UVA en la versión *July 2014*.

DISTRIBUCIÓN DE LA WEB

La Figura 1 muestra cómo está distribuida la web y cuáles son las partes más relevantes.

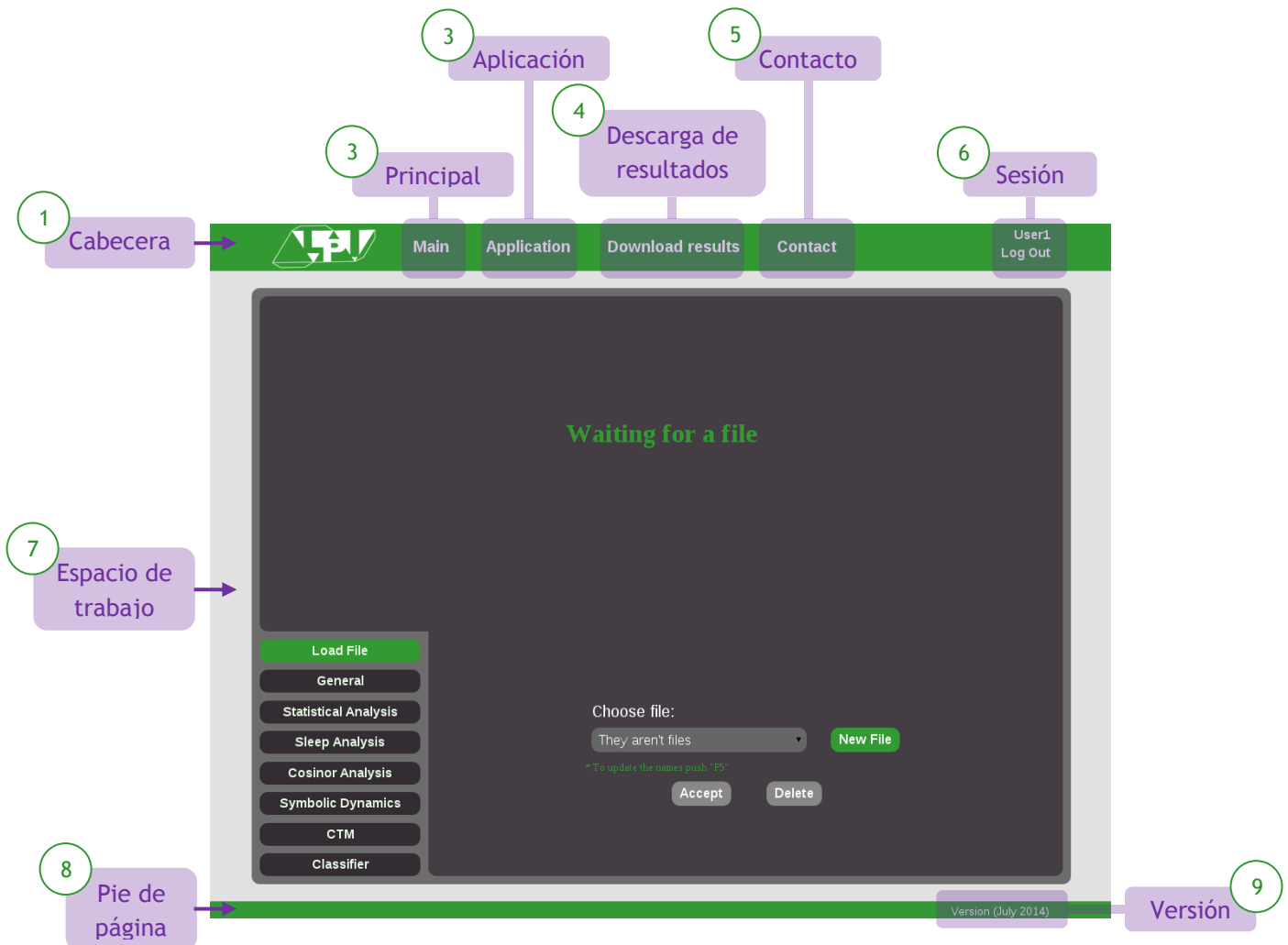


FIGURA 1

Cada una de las partes son enumeradas a continuación:

- 1) **Cabecera** : Desde este lugar puede accederse a cada una de las diferentes aplicaciones de la web.
- 2) **Principal** : Contiene una breve descripción de la web y el manual de usuario para la misma.
- 3) **Aplicación** : Este link da acceso a la aplicación más relevante, donde se podrán gestionar los registros y realizar cálculos sobre estos.
- 4) **Descarga de resultados** : Aquí se encuentra la aplicación que permite una mayor agilidad a la hora de procesar y obtener las características de los archivos actigráficos.
- 5) **Contacto** : Es un pequeño formulario que permite a cualquier usuario ponerse en contacto con nosotros.
- 6) **Sesión** : Permite el inicio y final de sesión de los usuarios registrados.
- 7) **Espacio de trabajo** : Es el espacio donde se mostrarán las aplicaciones y herramientas descritas.
- 8) **Pie de página** : Sirve para delimitar el final de la página
- 9) **Versión** : Muestra la versión actual la Web.

SISTEMA DE SESIONES

Para acceder a las aplicaciones es necesario tener una cuenta previamente creada. Lo sentimos mucho, pero la versión actual de nuestra web no permite aún la creación automática de una cuenta, para adquirir una, es necesario que se ponga en contacto con nosotros. Puede hacerlo desde <http://actigraphy.lpi.tel.uva.es/contact/>

INICIO Y CIERRE DE SESIÓN

La forma más sencilla de iniciar sesión es pulsando en la parte remarcada en la Figura 2, donde pone "Log in".

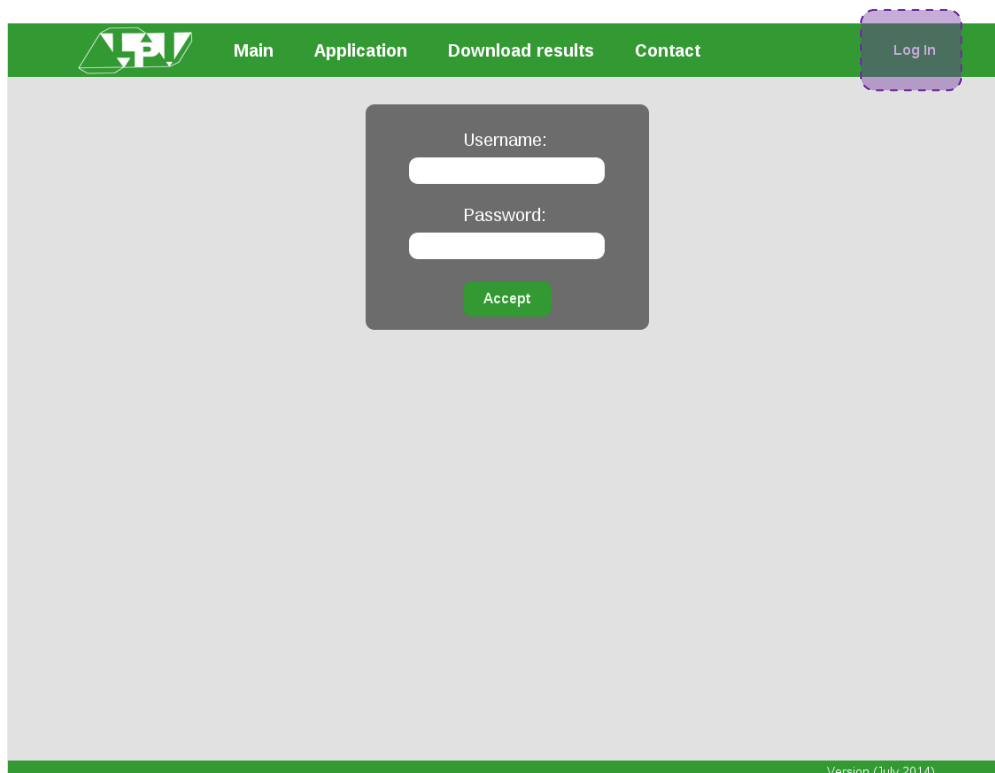


FIGURA 2

Ahora, el contenido mostrado debe ser similar al de la Figura 2. En este punto sólo queda rellenar los datos con los que le habremos proporcionado y pulsar el botón "Accept".

Una vez iniciada la sesión, el contenido de la cabecera pasará a mostrar el nombre del usuario, lo cual indica el inicio satisfactorio de la sesión. Ver Figura 3.

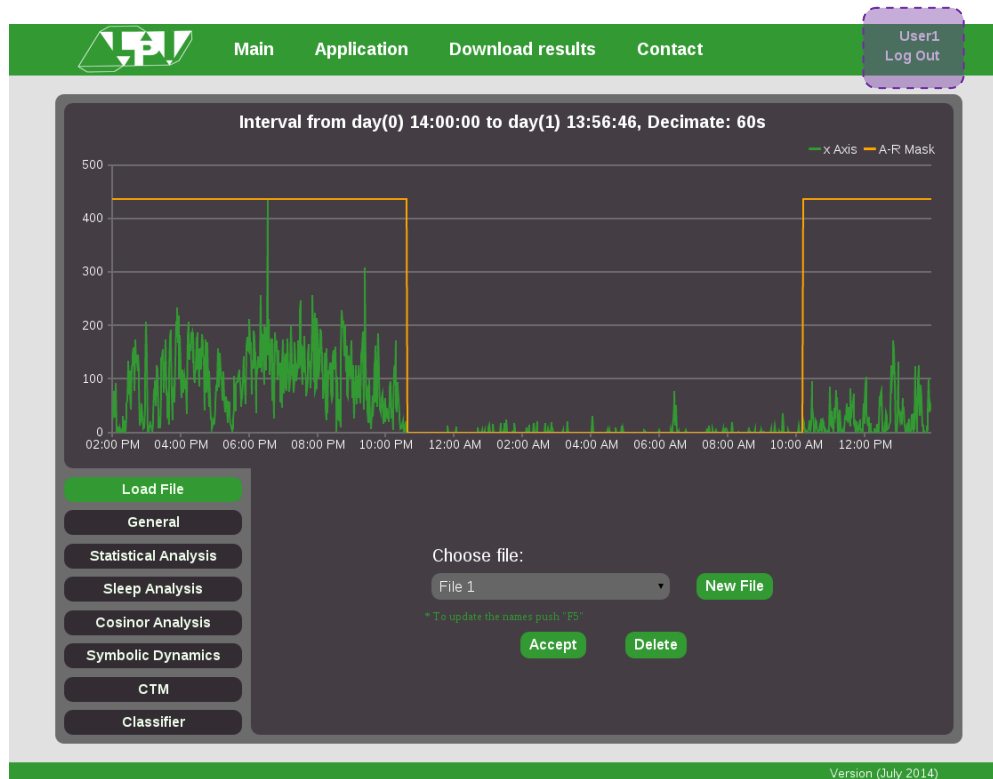


FIGURA 3

Como puede imaginarse, para cerrar sesión sólo debe pulsar el botón recientemente aparecido, nombrado como "Log Out".

APLICACIÓN

Si usted está ya registrado, podrá acceder directamente a nuestra aplicación, en el caso contrario, por favor lea el apartado *Inicio y cierre de sesión*.

ESTRUCTURA DE LA APLICACIÓN

La aplicación está dividida en las partes que muestran la Figura 4.

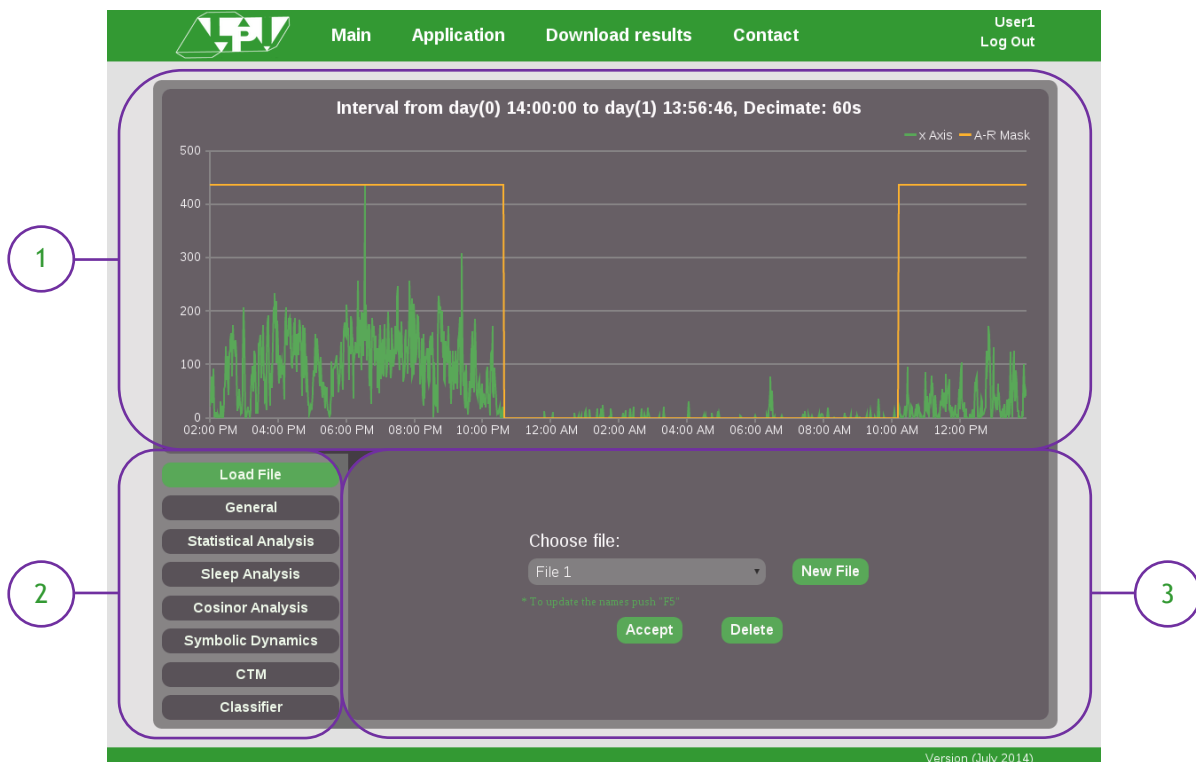


FIGURA 4

- 1) **Zona de resultados** : En esta zona se mostrarán tanto los resultados numéricos como las gráficas obtenidas.
- 2) **Zona de botones** : Muestra todas las herramientas implementadas en la aplicación.
- 3) **Zona de formularios** : Mostrará las diferentes posibilidades que nos permite cada una de las herramientas, ya sea para el cálculo de características, la administración de los archivos o la toma de notas.

LOAD FILE

Cuando se pulse el botón "Load File" la información mostrada en la zona de formularios variará, si es la primera vez que se accede a la aplicación, el contenido que debería ser similar al de la Figura 5. En ella se aprecian 3 botones: "Accept", "Delete" y "New File". También aparece un menú desplegable con la etiqueta "Choose file:".

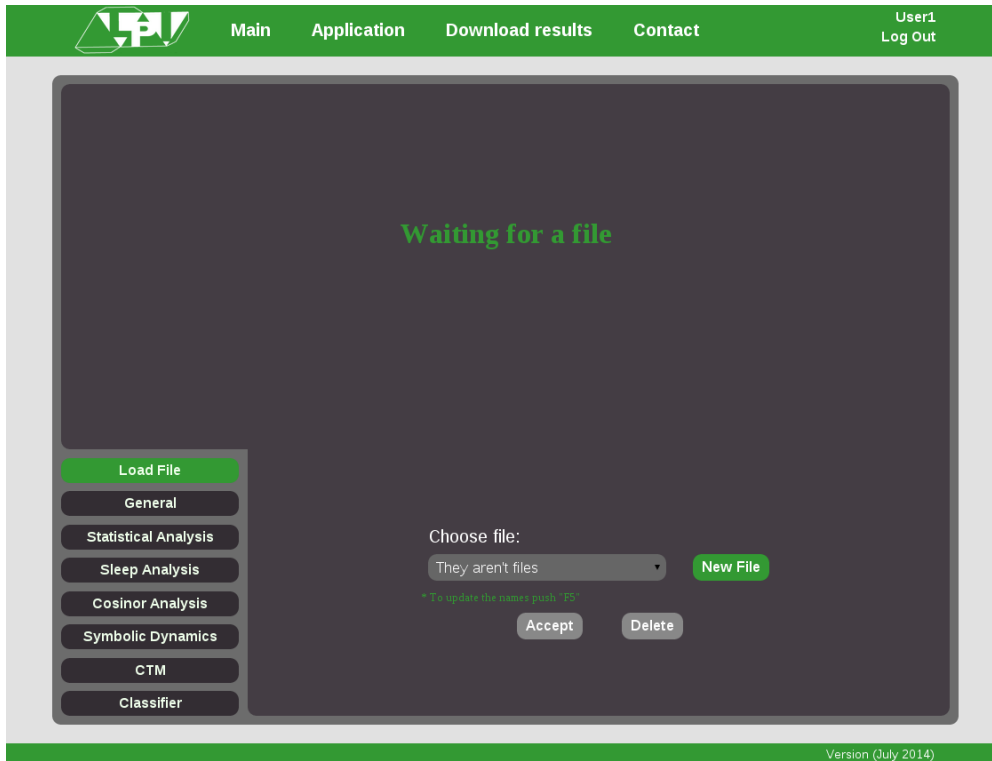


FIGURA 5

Como aún no existe ningún archivo, los botones "Accept" y "Delete" están deshabilitados y el menú desplegable indica que no hay archivos.

SUBIR UN NUEVO ARCHIVO

Para subir un archivo debe pulsarse en el botón "New file" visto en el Figura 5, esto nos dará acceso a un nuevo formulario (Ver Figura 6).

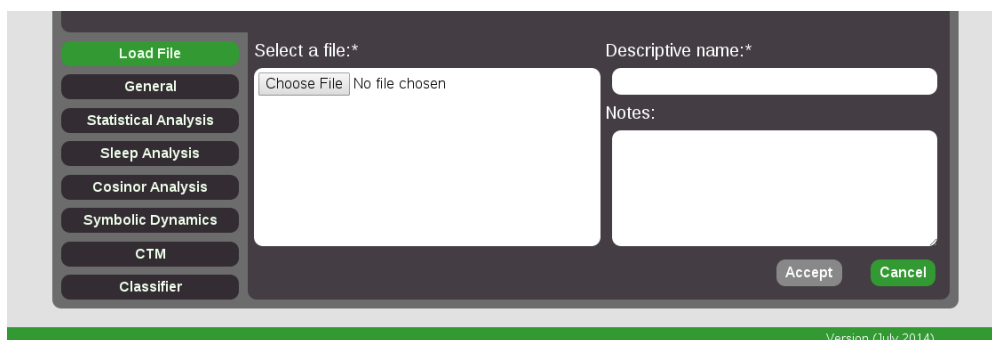


FIGURA 6

Aparecen 3 nuevas áreas:

- "Select a file" : Permite elegir o dejar caer un único archivo en él.
- "Descriptive name" : Es el nombre por el que se reconocerá el archivo dentro de la aplicación.
- "Notes" : Es un área opcional, en la cual el usuario puede anotar lo que desee que sea relevante para el archivo subido.

Una vez rellenados el área "Select a file" y "Descriptive name" correctamente, el botón "Accept" se habilitará, permitiendo acabar la acción de subir un archivo.

En la Figura 7 puede verse un ejemplo de cómo debe cumplimentarse este formulario.

FIGURA 7

Si el archivo no es adecuado o el nombre es demasiado largo, el servidor retornará un error con el tipo de excepción cometida.

Si todo funcionó correctamente, la aplicación procesará los datos del archivo automáticamente dando un acceso directo a toda la información del mismo como puede verse si nos fijamos en la zona de resultados de la Figura 8.

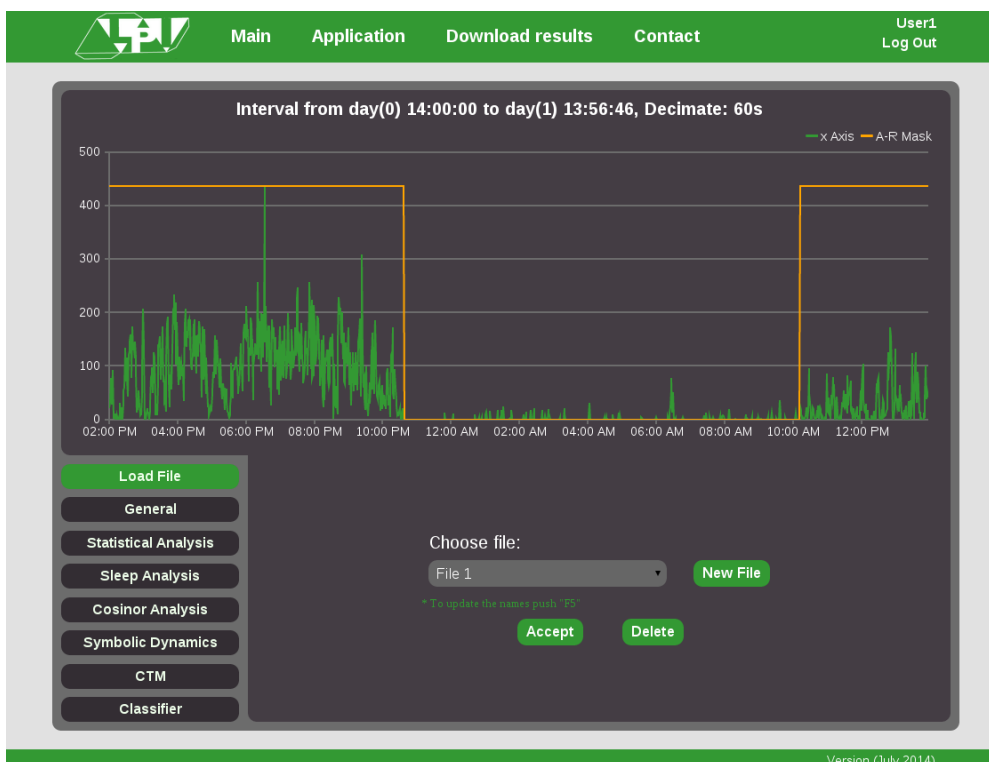


FIGURA 8

En la gráfica puede verse la información del intervalo y el diezmado actual por defecto que es de 60 segundos. Esta configuración puede modificarse en el apartado de General.

Cuando la información se está procesando, también se realiza el cálculo de la máscara de actividad-descanso que será la utilizada inicialmente en la aplicación. El algoritmo para su cálculo puede verse en el artículo: **D. Martín, P. Casaseca, S. Alberola, J.A. López, F.C. Ruiz, J.M. Andrés, J.R. Garmendia and J. Ardura. Automatic Diagnosis of ADHD Based on Nonlinear Analysis of Actimetry Registries. In 36th International Conference on Acoustics, Speech and Signal Processing (ICASSP'2011), 2011.**

También debe notarse que los botones "Accept" y "Delete" aparecen ya habilitados. El botón "Accept" sirve para elegir el archivo seleccionado en el menú desplegable, "Delete" lo elimina. **¡Atención!** *El archivo que haya sido eliminado no puede volver a recuperarse.*

GENERAL

Esta herramienta se activará cuando se pulse en el botón "General". Su función es la de modificar las características del registro actual para su posterior uso en otras herramientas.

En la Figura 9 puede verse cómo debería mostrarse la web en este punto.

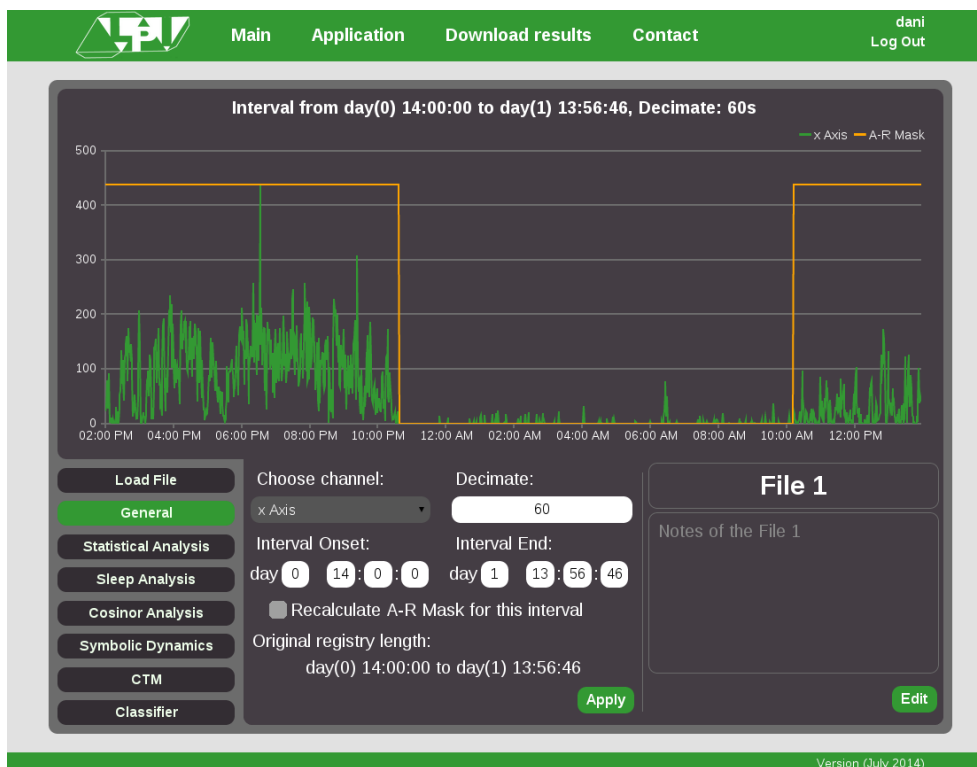


FIGURA 9

Aquí se plantean distintas opciones:

- **"Choose channel"**: Es un menú desplegable que permite elegir el canal sobre el que se desea trabajar.
- **"Decimate"**: Permite elegir el nivel de diezmado de la señal. Este nunca podrá ser inferior al nivel de detalle con el que se ha obtenido la misma.
- **"Interval Onset"**: Permite elegir el comienzo del intervalo sobre el cuál se trabajará.
- **"Interval End"**: Permite elegir el final del intervalo.
- **"Original registry length"**: Muestra los límites del intervalo original para que puedan mantenerse como referencia.
- **"Recalculate A-R Mask for this interval"**: Da la opción de recalculer la máscara de actividad descanso, el intervalo adecuado para calcular esta máscara debe ser de aproximadamente 24 horas. Lo ideal sería que el paciente comenzara su uso del actígrafo cuando se le proporciona, y lo siga utilizando hasta el momento de devolverlo.

GRÁFICA DEL CANAL Y LA MÁSCARA A-D

La gráfica de la Figura 10 muestra los datos con los que se está trabajando actualmente, los cuales serán utilizados en el resto de las herramientas.

Debido a la condensación de datos, estas gráficas aparecen sobrecargadas y su visualización resulta difícil. Como solución, Actigraphy LPI cuenta un sistema de zoom.

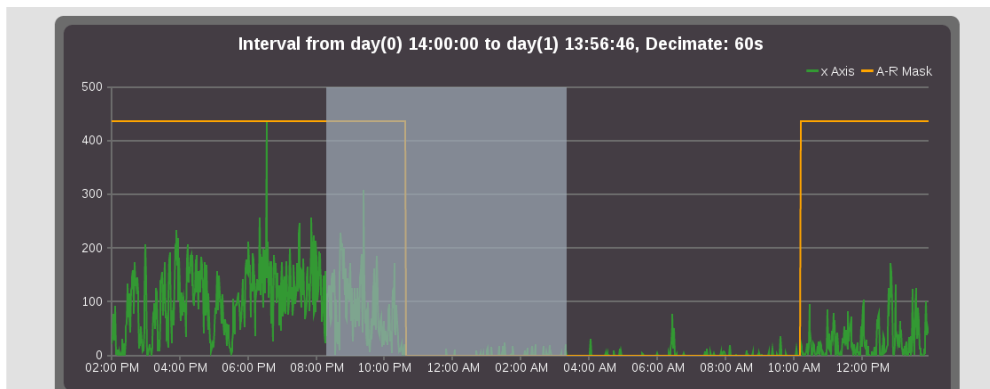


FIGURA 10

Este sistema se activa cuando selecciona una zona del gráfico con suficiente número de datos, como puede verse en la Figura 10. Una vez levantado el clic del ratón, se visualiza una ampliación del intervalo (Figura 11) (**¡Atención!** Esta opción sólo es útil para la visualización, no afecta a los datos con los que se trabaja).

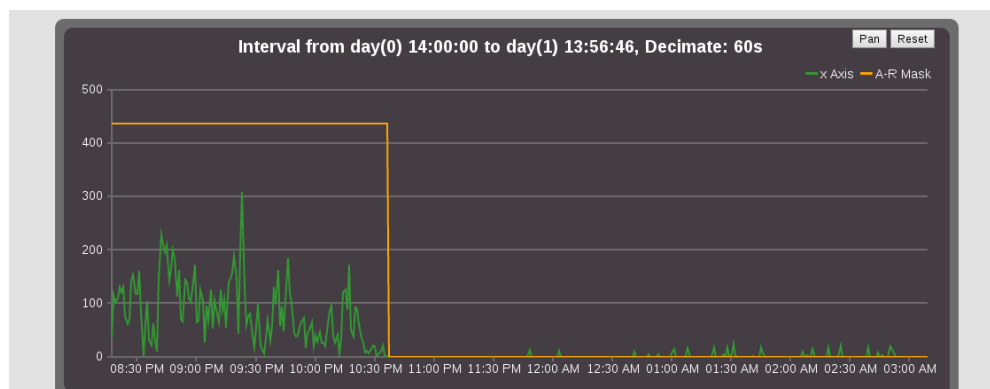


FIGURA 11

Al realizar esta acción, también aparecen dos botones en la parte superior derecha del gráfico:

- "Pan" : Permite hacer un scroll por los datos de la gráfica.
- "Reset" : Restaura la gráfica a su estado original.

NOTAS

En la parte inferior derecha de la Figura 9 hay dos campos, los cuales corresponden al nombre descriptivo y las notas. Estos pueden editarse en cualquier momento, basta con pulsar el botón "edit".

STATISTICA ANALYSIS

Esta herramienta permite calcular las características estadísticas del actual registro. Como puede verse en Figura 12, el formulario presenta dos opciones:

- "Frame": Selecciona qué parte del registro va a ser usada:
 - o "Activity": Utiliza la parte del registro en la cuál la máscara sea mayor que cero.
 - o "Rest": Utiliza la parte del registro en la cuál la máscara sea cero
 - o "Whole": Utiliza el registro completo
- "Number of bins": Indica el número de barras que se desea que tenga el histograma que va a calcularse.

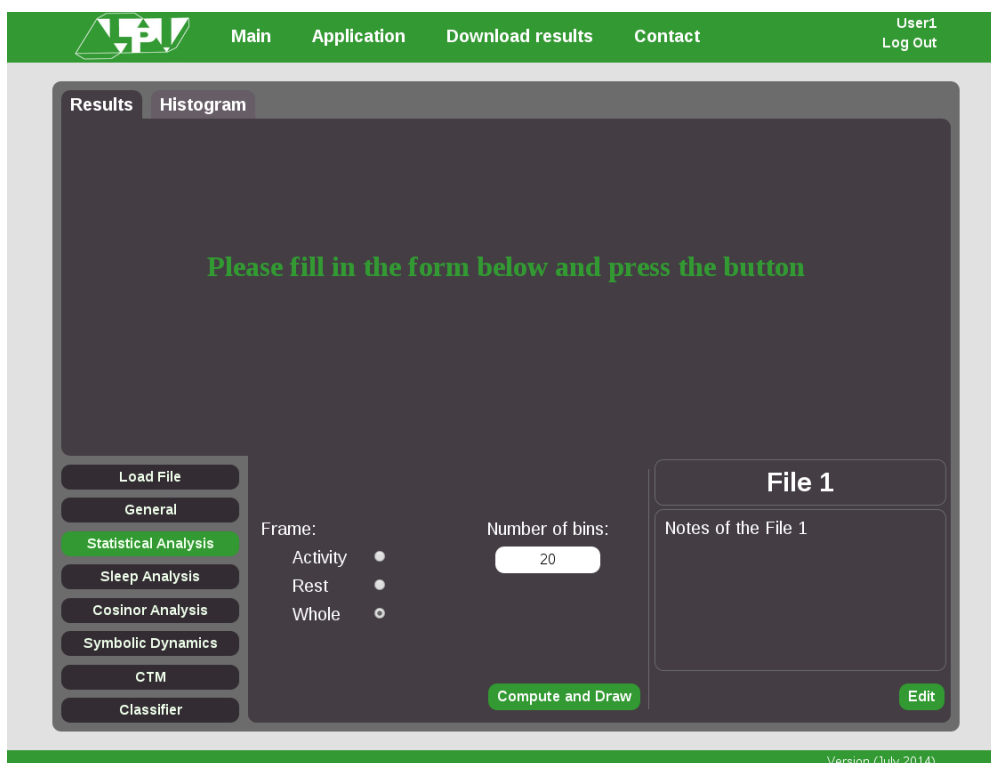


FIGURA 12

Una vez pulsado el botón "Compute and Draw", el resultado se mostrará mediante dos pestañas situadas en la parte superior izquierda de la Figura 12.

Los resultados se exponen de manera similar a la Figura 13. Los cálculos realizados son:

- "Mean" : Media aritmética.
- "Median" : Mediana.
- "Standard desviation" : Desviación estándar.
- "IQR" : Rango intercuartílico.
- "Skewness" : Coeficiente de asimetría.
- "Kurtosis" : Coeficiente de curtosis.
- "Variation cefficient" : Coeficiente de variación.

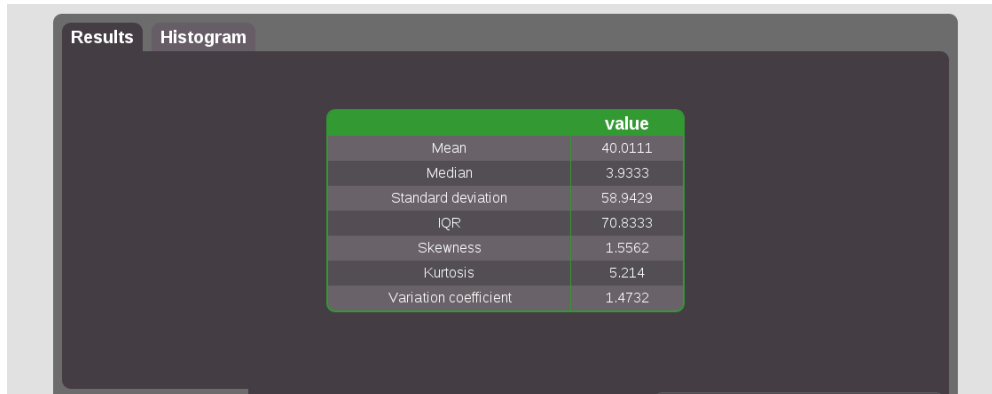


FIGURA 13

La Figura 14 muestra el cálculo del **histograma** de la señal normalizado. El número de barra se corresponde con el rellenado en el formulario. Al situar el ratón encima de cada una de las barras, se visualizará su valor central, es decir, donde se sitúa la barra en el eje x, así como el valor normalizado de la misma, el eje y.

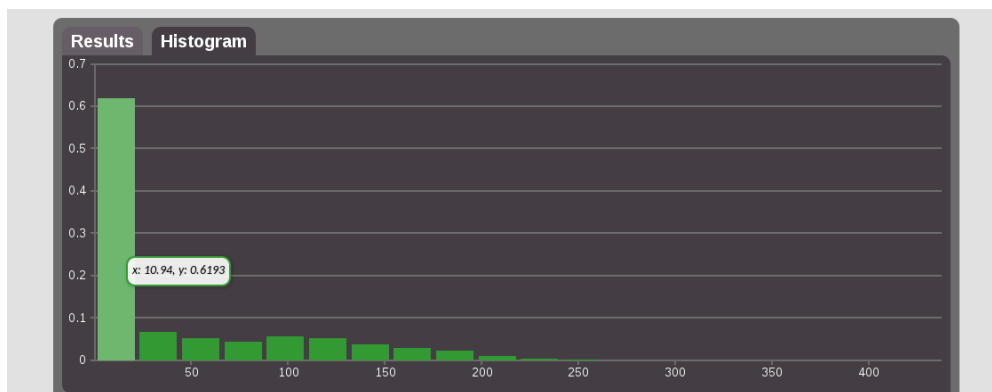


FIGURA 14

SLEEP ANALYSIS

Esta herramienta calcula las medidas características del sueño utilizando el algoritmo de Sadeh. Los datos que toma para el cálculo es el canal actual pero sin diezmar. El intervalo por defecto depende de la máscara de actividad - sueño, para cambiarlo, debe elegirse el modo manual (Figura 15).

FIGURA 15

Una vez pulsado el botón "Run Sadeh's Sleep Scoring", se mostrarán los resultados en la parte superior de la aplicación (Figura 16).

Sleep quality parameter	Scoring
Sleep Onset (min)	6
Total Time of Sleep (min)	611
Wake After Sleep Onset (min)	79
Awakenings	34
Average Awakening (min)	2.32352940493
Total Counts	147396.2
Efficiency (%)	87.2857142857
Longest Sleep Interval (min)	85.0
Percent Quite Sleep (%)	88.5507246377

FIGURA 16

COSINOR ANALYSIS

Este método sigue el esquema establecido por Nelson W, Tong YL, Lee JK, Halberg F. *Methods for cosinor rhythmometry. Chronobiologia.1979;6:305-323* donde se ajusta una curva sinusoidal a la señal actual mediante el método de los mínimos cuadrados. El formulario cuenta con dos opciones (Figura 17):

- "Rhythm" : Establece el periodo al cual debe ajustarse el cosinor.
- "Alpha" : Es el margen usado para el cálculo del intervalo de confianza.

La señal que se utiliza para calcular el cosinor es la establecida en el apartado General.

FIGURA 17

Al ejecutar el programa, se devuelven los resultados y dos gráficas. En el apartado de resultados, se ven las principales características de la onda calculada y los intervalos de confianza (Figura 18).

	Estimated Value	CI: Inferior Limit	CI: Superior Limit
Mesor	40.0583387968932	0	89.8147526414210
Amplitude	59.0080782581	56.3438372730831	61.6747726435483
Acrophase	-17.9060045102	-18.0804576841	-17.7315739968
p-value	8.73855140417e-218	0	0

FIGURA 18

La primera gráfica "Cosine fit", es la onda calculada sobrepuesta a la señal original (Figura 19).

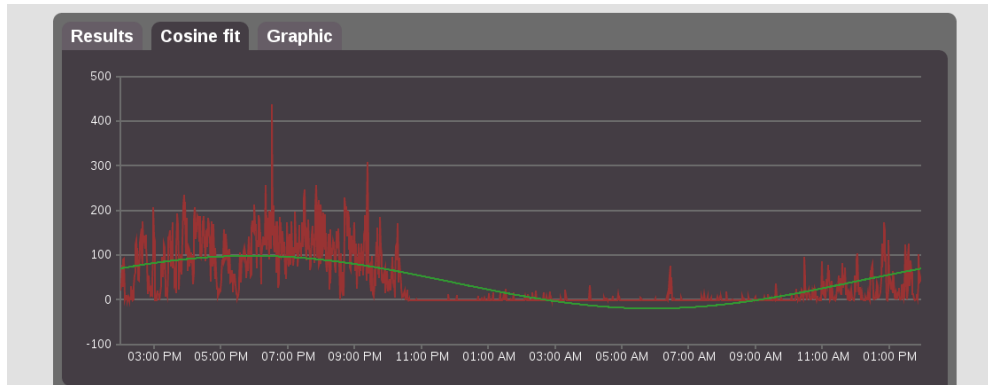


FIGURA 19

Finalmente en el apartado "Graphic" (Figura 20) se dibuja un resumen de forma gráfica donde:

- **Vector**: Es un vector definido por la "Amplitude" y la "Acrophase".
- **Elipse**: Es la elipse de confianza.

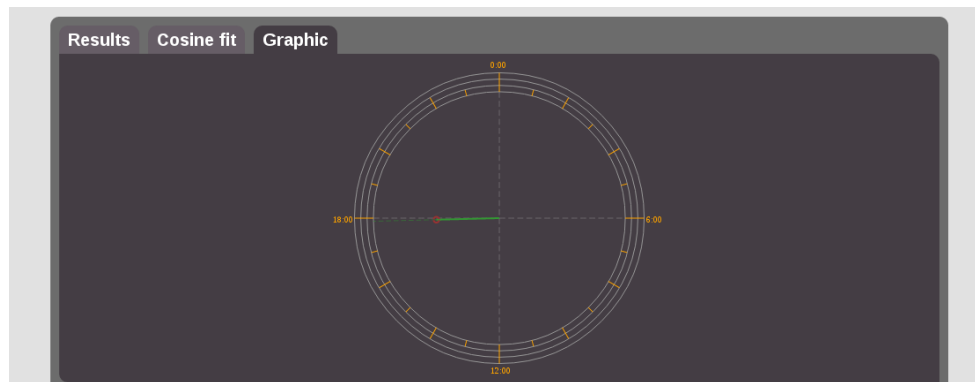


FIGURA 20

SYMBOLIC DYNAMICS

Para esta herramienta se aplica el algoritmo utilizado en D. Martín-Martínez, P. Casaseca-de-la-Higuera, J. Ardura-Fernández and C. Alberola-López. NonlinearAnalysis of Actigraphic Signals for the Assessment of the Attention-Deficit/Hyperactivity Disorder (ADHD)" Medical Engineering & Physics, vol. 34, pp. 1317-1329, 2012.

Las opciones para este cálculo son (Figura 21):

- "Frame": Selecciona qué parte del registro va a ser usada:
 - o "Activity": Utiliza la parte del registro en la cuál la máscara sea mayor que cero.
 - o "Rest": Utiliza la parte del registro en la cuál la máscara sea cero
 - o "Whole": Utiliza el registro completo
- "Alphabet length" : Longitud del alfabeto.
- "Symbols / Word" : Número de símbolos por cada palabra.
- "Overlapping" : Solapamiento entre las palabras. Un solapamiento de 0 significaría no existiría solapamiento alguno, mientras que si su valor es de (Symbols/Word-1) el solapamiento sería máximo.

FIGURA 21

Los resultado deben mostrarse de forma parecida a la expuesta en la Figura 22.

Word	Probability
00	0.376306620209
01	0.0543554006969
02	0.000696864111498
10	0.0543554006969
11	0.14425087108
12	0.0376306620209
20	0.000696864111498
21	0.0369337979094

FIGURA 22

CTM (CENTRAL TENDENCY MEASURE)

Da la medida de la tendencia central de la señal. El algoritmo utilizado en D. Martín-Martínez, P. Casaseca-de-la-Higuera, J. Ardura-Fernández and C. Alberola-López. NonlinearAnalysis of Actigraphic Signals for the Assessment of the Attention-Decit/Hyperactivity Disorder (ADHD)" Medical Engineering & Physics, vol. 34, pp. 1317-1329, 2012.

Las opciones de esta herramienta (Figura 23) permiten fijar :

- "Frame": Selecciona qué parte del registro va a ser usada:
 - o "Activity": Utiliza la parte del registro en la cuál la máscara sea mayor que cero.
 - o "Rest": Utiliza la parte del registro en la cuál la máscara sea cero
 - o "Whole": Utiliza el registro completo
- "Rho" : Es el radio del círculo donde se estima la concurrencia de la señal.

The screenshot shows the Actigraphy LPI software interface. At the top, there is a green navigation bar with the LPI logo and links for 'Main', 'Application', 'Download results', and 'Contact'. On the right side of the bar, it shows 'User1' and a 'Log Out' button. The main content area has a dark background with the text 'Please fill in the form below and press the button' in green. On the left, there is a vertical menu with buttons for 'Load File', 'General', 'Statistical Analysis', 'Sleep Analysis', 'Cosinor Analysis', 'Symbolic Dynamics', 'CTM' (highlighted in green), and 'Classifier'. The central form area contains a 'Frame:' label with three radio button options: 'Activity', 'Rest', and 'Whole'. To the right of these is a 'Rho:' label with a text input field containing the value '0'. Below the 'Rho' field is a green 'Compute' button. On the right side of the form, there is a section titled 'File 1' with a 'Notes of the File 1' text area and an 'Edit' button. At the bottom right of the interface, it says 'Version (July 2014)'.

FIGURA 23

Una vez establecidos, puede pulsar el botón "Compute", obteniendo una gráfica parecida a la Figura 24. En su encabezamiento se encuentra el valor calculado, así como el valor de rho.

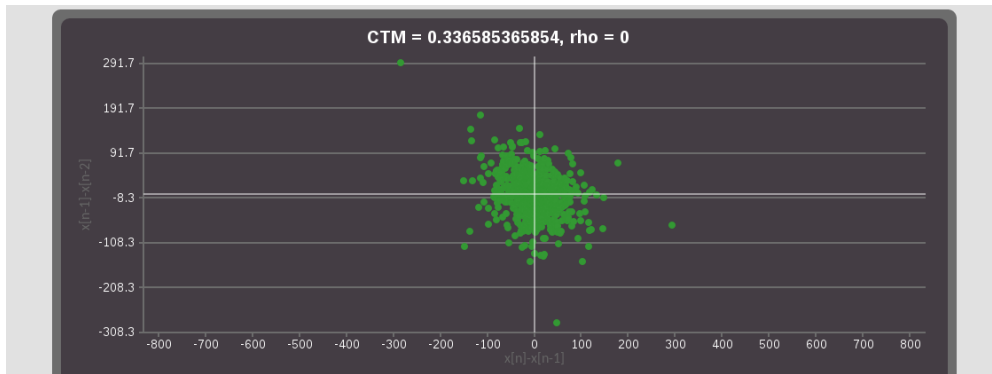


FIGURA 24

CLASSIFER

Para esta herramienta permite clasificar el registro actigráfico para saber si un paciente con edad cercana a 6 años padece TDAH o no:

- "Interval Onset": Permite elegir el comienzo del intervalo sobre el cuál se trabajará.
- "Interval End": Permite elegir el final del intervalo.
- "Choose classifier": Es un menú desplegable que permite seleccionar el tipo de clasificador que se desea utilizar. La actual versión de la aplicación sólo cuenta con el clasificador K-NN
- "K" : Es una opción propia del tipo de clasificador, en este caso pertenece al K-NN y permite elegir el número de vecinos con el que desea realizarse.

FIGURA 25

Los resultado deben mostrarse de forma parecida a la expuesta en la Figura 26.

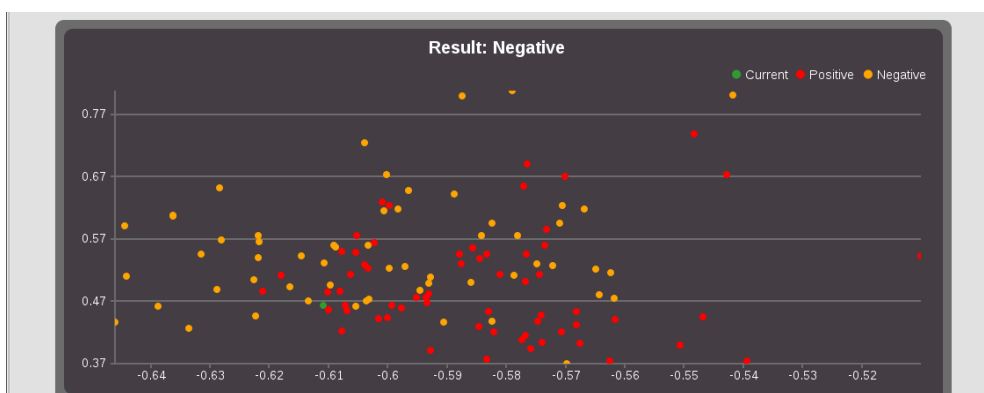
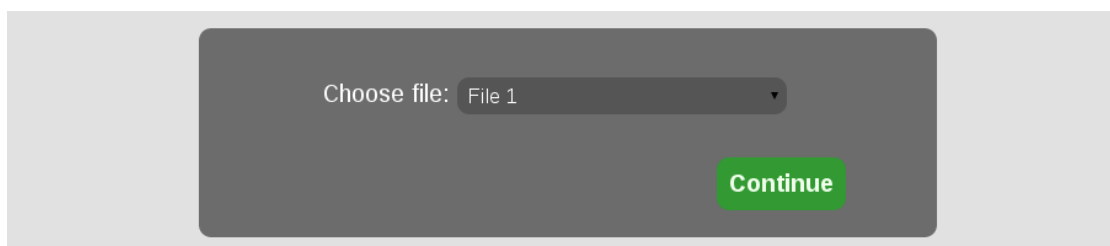


FIGURA 26

DESCARGA DE RESULTADOS

Actigraphy LPI proporciona una utilidad capaz de generar una hoja de cálculo que contiene todos los datos que pueden calcularse con nuestra aplicación pero en sólo tres pasos:

Seleccione cualquier archivo actigráfico que haya subido con anterioridad y pulse el botón "Continue" (Figura 27)

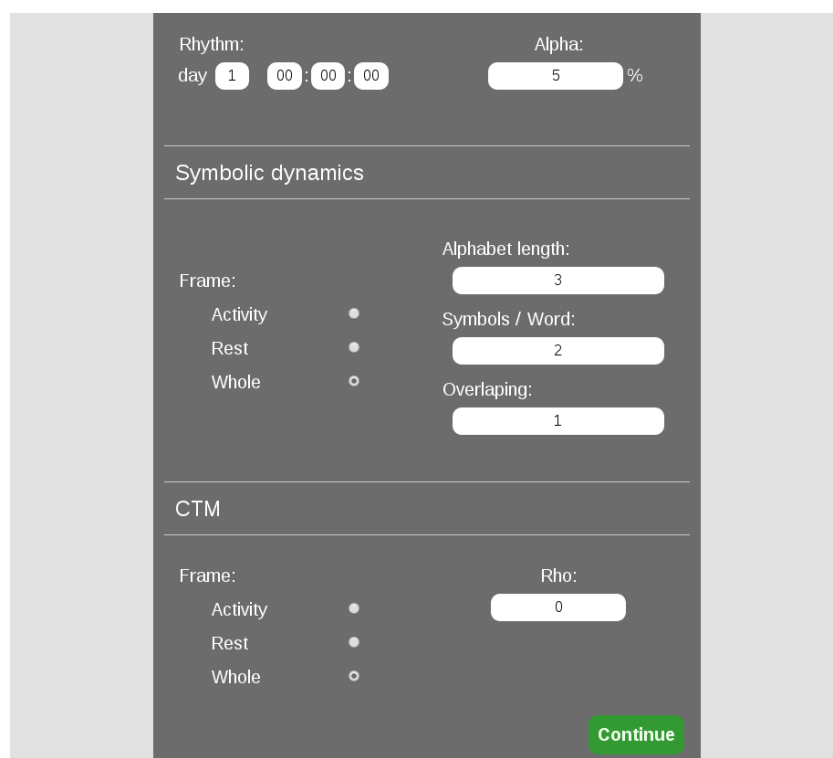


Choose file: File 1

Continue

FIGURA 27

Rellene el formulario como usted crea conveniente y vuelva a pulsar el botón "Continue" (Figura 28)



Rhythm: day 1 00 : 00 : 00 Alpha: 5 %

Symbolic dynamics

Frame: Alphabet length: 3

Activity • Symbols / Word: 2

Rest • Overlapping: 1

Whole ○

CTM

Frame: Rho: 0

Activity •

Rest •

Whole ○

Continue

FIGURA 28

Nuestra aplicación comenzará a realizar el cálculo de forma automática, cuando el proceso finalice sólo debe descargar el fichero.

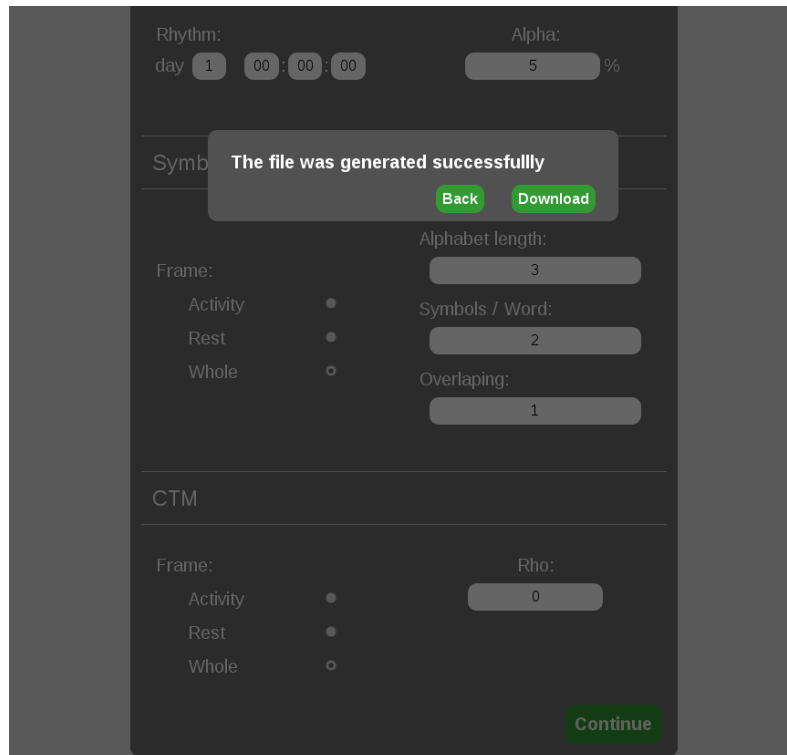


FIGURA 29

Las herramientas son las mismas descritas en el apartado de Aplicación. Si usted tiene alguna duda de lo que plantean las opciones, por favor, no dude en preguntárnoslo.

